

参赛密码 _____
(由组委会填写)



“华为杯”第十四届中国研究生 数学建模竞赛

学 校 青岛科技大学

参赛队号 10426002

队员姓名 1.贾福浩

2.游 迪

3.安 泰

参赛密码 _____
(由组委会填写)



“华为杯”第十四届中国研究生 数学建模竞赛

题 目 车载导弹波次作战部署

摘 要：

本文研究了车载导弹波次运输规划问题。在高技术作战中，需要对车载导弹运输进行作战规划，将多枚导弹运输到指定的发射位点实施波次作战，这在双方博弈中发挥着极其重要的作用，直接关系着作战的效果。我们结合作战区域分布特点及发射装置速度特点，建立了 0-1 规划模型，并使用改进的 Dijkstra 算法求解最短暴露时间，进行导弹运输的组合优化。

问题一是连续两个波次的导弹运输规划问题。其特点是：不同发射装置在不同道路上的速度不是固定值，属于动态规划。首先，我们以时间为权值构建时间依赖网络，将求最短距离的 Dijkstra 算法转变为求最短时间的 Dijkstra 算法，作出时间矩阵。将两个波次的运输规划分为两个阶段的组合优化：(1) 第一阶段为从待机区域到第一波次的发射点位；(2) 第二阶段为从第一波次发射位点前往转载地域装弹后再机动到第二波次的发射点位。对这两个阶段，根据时间矩阵建立了一个 0-1 规划模型确定第二阶段的路径。然后考虑单行道上发生超车的节点等待时间，装弹完毕后两车在单行道上出现交会的节点等待时间及在转载地域的等待暴露时间，对 0-1 规划模型进行补充修正，采用了先到先服务 (FCFS) 等待制的排队规则进行排队，输出值为包含有等待时间的整体暴露时间。由此我们得到两波次的运输方案的整体暴露时间为 9849.7min。

问题二和问题三分别是两个临时转载地域的部署和选取新增发射装置的隐蔽点位。临时转载地域的部署可能的方案有 $C_5^2 = 10$ 种，选取隐蔽点位的可能方案有 $C_6^3 + A_6^2 = 50$ 种。并且临时转载地域的部署和隐蔽待机点的选择均不影响第一阶段的运输规划。第二阶段采用了第一问建立的修正 0-1 规划模型，对每种方案进行模拟，求解第二阶段暴露时间。通过排序比较，得到临时装载地域选取在 J34、J25 时，实施两个波次发射任务的整体暴露时

间最短，为8465.1min。问题三中，基于问题一中第二阶段的单台 C 类发射装置的暴露时间，筛除了 C07, C03, C12 发射装置。然后构建不同方案的时间矩阵，通过问题一的模型，求出各方案的整体暴露时间，通过排序比较，得到选取 J06, J14, J15 作为隐蔽点位时，实施两个波次发射的整体暴露时间最短，为 8902.6min。

问题四是通过对道路节点重要性的评价，得到容易被敌方损毁的道路节点。我们采用 TOPSIS 综合评价方法建立了道路节点重要性的评价模型，重要性指数越高越容易被敌方损毁。根据该作战区域的网络结构特点，用度指标、介数指标、紧密度指标对道路节点重要性进行综合定量分析，最后得到容易被损毁的节点是 J09, J15, J55。

问题五是考虑实际作战多种因素，对问题一的运输路径进行优化。除保证整体暴露时间最短外，我们将运输任务尽量分散来提高隐蔽性和可靠性。用运输流量的信息熵来表示隐蔽性，建立了一个随机规划模型，以信息熵为目标函数，用车辆运行时间和道路容量进行约束，得到优化结果。

关键词：时间依赖；0-1 规划；Dijkstra 算法；排队论；TOPSIS；随机规划。

目录

1. 问题重述	5
1.1 问题背景	5
1.2 需解决的问题	6
2. 模型假设	7
3. 通用符号说明	7
4. 问题一：两个波次的发射任务规划	7
4.1 问题分析	7
4.2 数据处理	9
4.3 模型建立	9
4.3.1 时间邻接矩阵建立	9
4.3.2 最短时间的 Dijkstra 算法过程	10
4.3.3 0-1 规划模型	11
4.3.4 0-1 规划修正	12
4.4 模型求解	13
4.4.1 第一阶段 Dijkstra 算法求解	13
4.4.2 0-1 规划求解	15
4.4.3 第二阶段修正	17
4.5 结果分析	17
5. 问题二：临时转载地域的部署	19
5.1 问题分析	19
5.2 模型准备	19
5.3 模型建立及求解	19
5.3.1 某一方案的整体暴露时间	19
5.3.1 选取最优方案	21
5.4 结果分析	21
6. 问题三：新增发射装置的隐蔽位点选取	22
6.1 问题分析	22
6.2 模型准备	23
6.3 模型建立	24
6.3.1 新增发射装置的时间矩阵	24
6.3.2 第二阶段 0-1 规划	24
6.3.3 确定最优隐蔽点位选取方案	24
6.3 模型求解	25
6.3.1 第一波次后发射装置的筛选	25
6.3.2 新增发射装置后的暴露时间求解	25
6.3.4 比较法确定确定隐蔽点位选取方案	26
6.4 结果分析	26
7. 问题四：道路节点损毁的评价体系	27
7.1 问题分析	27
7.2 三个评价指标的定义	27
7.3 模型建立	27
7.3.1 三个指标归一化处理	27
7.3.2 TOPSIS 综合评价法	28
7.4 模型求解	28
7.4 结果分析	29
8. 问题五：两个波次发射任务的优化	30
8.1 问题分析	30
8.2 模型建立	30
9. 模型评价及优化	31

9.1 模型优点及推广	31
9.2 模型缺点及改进	31
参考文献.....	34
附录.....	34

1. 问题重述

1.1 问题背景

车载导弹具有射程远、威力大、精度高、打击范围宽的特点，是一种新型的、综合性作战武器，在高技术作战中发挥着极其重要的作用。

采用车载发射装置发射导弹，发射装置平时部署在待机区域隐蔽待机，在接受发射任务后，携带导弹沿机动道路到指定的发射点位发射，一个发射点位最多容纳一台发射装置。在连续多波次的打击任务中，每个车载发射装置一次只能装载一枚导弹，完成一波发射后需要回到转载地域装弹，一个转载地域最多能容纳 2 台发射装置，但不能同时装弹，完成一次装弹作业需时 10 分钟。各转载地域弹种类型和数量满足需求。完成装弹后再沿机动道路到下一个指定的发射点位发射，且连续两波发射中同一发射点位只能使用一次。

某部参与作战的车载发射装置可分为 A、B、C 三类，共 24 台，其中 A、B、C 三类发射装置的数量分别为 6 台、6 台、12 台，执行任务前平均部署在 2 个待机地域（D1，D2），即每个待机地域 A、B、C 的数量分别是 3 台、3 台、6 台，共 12 台。所属作战区域内有 6 个转载地域（Z01~ Z06）、60 个发射点位（F01~ F60）。相关道路情况如图 1-1 所示，相关要素坐标如附件 1 所示。由于实际的路况限制，主干道路（图 1-1 中红线）是双车道，可以双车通行，其他支路（图 1-1 中蓝线）是单车道，只能在各道路节点会车。A 类发射装置在主干道和其他支路的速度分别为 70 公里/小时、40 公里/小时；B 类发射装置在主干道和其他支路的速度分别为 60 公里/小时、25 公里/小时；C 类发射装置在主干道和其他支路的速度分别为 50 公里/小时、30 公里/小时。

在接到发射任务后，需要规划一个发射任务的合理分配方案和机动方案，使整体暴露时间最短。暴露时间是指各车载发射装置从待机地域出发时刻至第二波次发射时刻为止的时间，其中发射装置位于转载地域内的时间不计入暴露时间内，不考虑在发射点位必要的技术准备时间，且在发射点位发射完毕后立即机动前往转载地域。

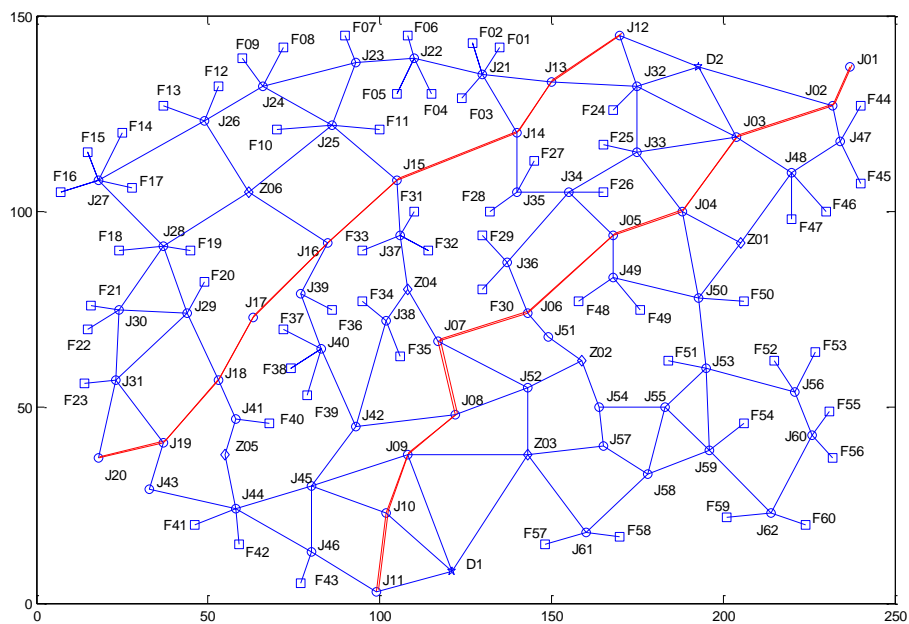


图 1-1 作战区域相关道路图

1.2 需解决的问题

(1) 现接受到两个波次的齐射打击任务，每个波次齐射 24 枚导弹，规划出一个发射任务的合理分配方案和机动方案，使得两个波次的整体暴露时间最短。

(2) 除已有的 6 个转载地域 (Z01~Z06) 外，可选择在道路节点 J25、J34、J36、J42、J49 附近临时增设 2 个临时转载地域，请对转载地域进行合理部署，确定临时转载地域布设坐标 (坐标选取相应节点的坐标) 使得两个波次的齐射打击任务的总体暴露时间最短。

(3) 现在第二波次发射过程中增加 3 台 C 类车载发射装置，3 台 C 类发射装置的隐蔽待机点位可在道路节点 J04、J06、J08、J13、J14、J15 中选取，一个隐蔽待机点位最多容纳 2 台车载发射装置，从待机地域到隐蔽待机点位的时间不计入暴露时间内。在第一波次发射完毕后，新增的 3 台 C 类车载发射装置从隐蔽待机点机动前往第二波次指定的发射点位，不参与第二波次的原 3 台 C 类发射装置完成第一波次的发射任务后返回待机区域，其返回时间不计入暴露时间。在转载地域为 6 个的情况下，确定新增发射装置的隐蔽待机点，使得完成两个波次打击任务的总体暴露时间最短。

(4) 考虑到战时双方博弈，会有道路打击损毁的情况出现，造成导弹运输过程中延迟甚至阻碍发射装置按时到达指定发射点位。结合图 1-1 的道路网络特点，建立合理的道路损毁评价体系，量化分析在该作战区域最可能受到敌方打击损毁的 3 个道路节点。

(5) 在实际作战中，提高发射装置运输隐蔽性是提高战前武器装备生存能力的重要途径，既要考虑使从待机区域到发射点位的整体暴露时间最短，也要考虑到运输任务尽量分散到各条线路的隐蔽性和可靠性，同时考虑使单台发射装置的暴露时间尽量短，综合以上因素，对问题 (1) 中两个连续波次的打击任务重新进行规划。

2. 模型假设

- 1) 假设发射装置在途中不会发生故障；
- 2) 假设敌方每个目标都在打击范围内；
- 3) 假设单车道不允许超车；
- 4) 假设转载地域除隐蔽装弹区域外其他区域可以视为节点，允许多辆发射装置在此处会车等待，但其等待时间仍计入暴露时间内；
- 5) 假设在转载地域装弹完毕后立即前往下一波发射点位；
- 6) 假设发射装置在运输途中始终以平均速度行驶；

3. 通用符号说明

序号	符号	符号说明
1	W_A	依据 A 类发射装置速度的时间邻接矩阵
2	W_B	依据 B 类发射装置速度的时间邻接矩阵
3	W_C	依据 C 类发射装置速度的时间邻接矩阵
4	n	第 n 个阶段
5	k	第 k 台发射装置
6	$v_{k,i}$	第 k 台发射装置机动路径的第 i 个节点
7	$t_{v_i^k v_{i+1}^k}^n$	第 k 辆车在第 n 阶段相邻两节点的机动时间
8	T^n	第 n 阶段的暴露时间

4. 问题一：两个波次的发射任务规划

4.1 问题分析

问题一是发射两个波次的车载导弹运输规划问题，可以简单的描述为：在 D1、D2 待机的 24 台发射装置装载导弹进行两个波次打击任务，规划每台发射装置在每个波次对应的发射点位及机动路线方案，使得整体暴露时间最短。

首先，该问题属于图与网络问题，由于 A、B、C 三类不同发射装置在不同的道路上速度不同，是一个动态规划问题，用传统的最短路方法无法得到最优解。我们考虑以时间为权值构建道路的时间依赖网络^[1]，具体思路为：根据 A、B、C 三类发射装置在道路上速度不同构建三个时间邻接矩阵，矩阵权值由对应两节点的距离与对应发射装置速度比值给出。由于两个波次要求齐射，所以可以将求解全局最优化简为求解两个局部最优，而两个阶段的最优值相加即为全局最优（如图 4-1）。将任务分为两个阶段，第一阶段为从待机区域 D1、D2 到第一个波次的

发射点位，第二阶段为第一个波次的发射点位前往装载地域（Z01~ Z06）装弹，然后从装载地域前往第二个波次的发射点位。分别对两个阶段进行分析，求暴露时间最短的最优解。

针对第一阶段，根据建立的时间邻接矩阵，拟采用改进的Dijkstra算法求解三类发射装置从待机区域D1、D2到发射点位（F01~ F60）时间最短的路径，并进行排序。考虑发射装置都具有在主干道上速度快，在其他道路上速度慢的特点，所以每类发射装置从D1、D2到60个发射点位时间排序应大致相同，选取到达待机区域D1、D2时间较短的前12个发射点位作为第一个波次的发射点位，验证分别选取的12个发射点位是否存在重复，若存在重复的情况，按时间顺序往下递推选取，直至分别选取的12个发射点位不存在重复为止。对于发射装置对应发射点位的分配问题，我们以暴露时间为目标函数，用0-1规划进行约束，求解使暴露时间最短的分配方案。

针对第二阶段，规划第一波次发射完毕后进行第二波发射的问题，前往第二波发射点位之前要去转载地域装弹。该阶段属于第一波发射点位至转载地域至第二波发射点位的优化问题，需要综合考量三类节点之间的路径^[2]。对某一发射装置起点位置是固定的，因为连续两波中发射点位最多使用一次，所以其终点有36种可能的发射点位，而对每一可能的发射点位，都有6个可选择的中停点（转载地域），我们拟采用第一阶段改进的Dijkstra算法，固定某一发射装置的起止点，求解经过6个中停点的最短时间路径，通过比较时间，选取经过中停点时间最短的路径作为该固定起止点的路径（如图4-2）。从而可以得到每一发射装置到36种可能得发射点位的最短路径及其中停点。然后以该阶段的暴露时间为目标函数，采用0-1规划对目标函数进行约束，求解最小值。确定机动方案后，对多辆车去往转载地域存在的等待暴露时间进行优化计算，得到第二阶段的整体经过各节点的总时间。

暴露时间还包括单行道上发生超车的节点等待时间，装弹完毕后两车在单行道上出现交会的节点等待时间，再转载地域的等待暴露时间。拟参考排队论的思想，采用等待制先到先服务（FCFS）的排队规则^[3]进行排队，输出等待时间。

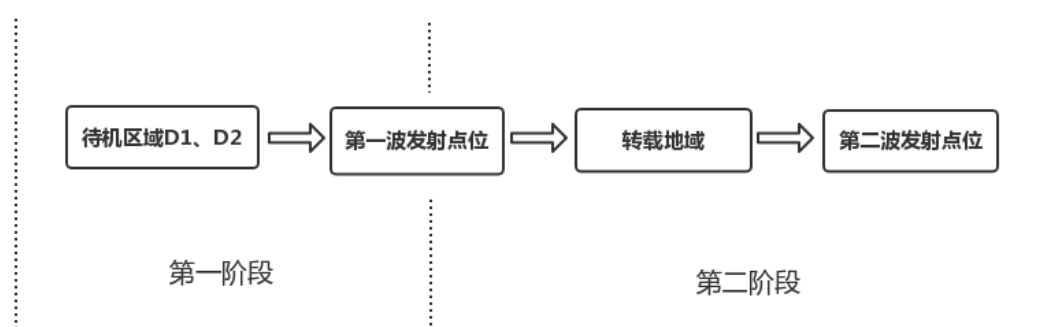


图4-1 两波次导弹规划运输

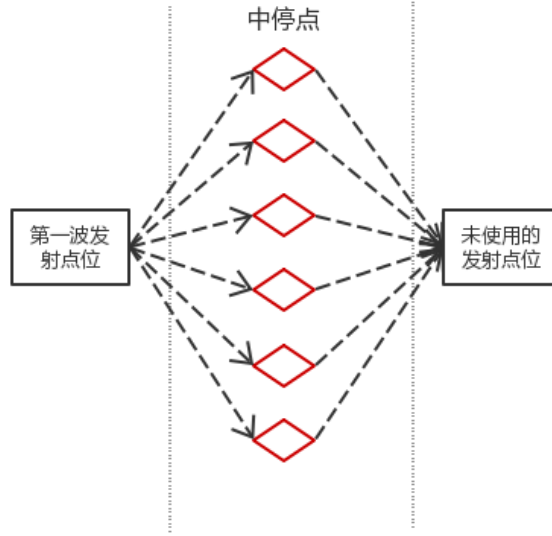


图4-2第二阶段分析示意图

4.2 数据处理

(1) 道路节点标记处理

为了在 MATLAB 中标记待机区域 D1、D2，转载地域 (Z01~Z06)，发射点位 (F01~F60)，道路节点 (J01~J62)，我们将各节点用纯数字标记，如表 4-1 所示。

表 4-1 道路节点标记

道路节点	D1-D2	Z01-Z06	F01-F60	J01-J62
数字标记	1-2	3-8	9-68	69-130

(2) 坐标数据处理

附件 1 中给出的是各节点的坐标数据，将坐标转化为连通道路的距离数据：

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

i, j 分别表示不同节点，当 i, j 之间无道路连接时用 “ ∞ ” 表示。

4.3 模型建立

4.3.1 时间邻接矩阵建立

邻接矩阵是表示顶点之间相邻关系的矩阵，本题中的网络具有时间依赖特点，所以对 A、B、C 三类车分别建立三个时间邻接矩阵 W_A 、 W_B 、 W_C 。

不失一般性，以 A 类车在发射装置为例构建时间邻接矩阵。作战区域共有 n 个节点，用 $W_A = (w_{ij})_{n \times n}$ 表示邻接矩阵，权值由对应两节点的距离与对应发射装置速度比值给出：

$$w_{ij} = \begin{cases} t_{ij}, & \text{当节点 } i \text{ 与 } j \text{ 之间有道路连通时} \\ \infty, & \text{当节点 } i \text{ 与 } j \text{ 之间无道路连通时} \end{cases} \quad (4-1)$$

其中，

$$t_{ij} = \frac{d_{ij}}{v_A} \quad (4-2)$$

4.3.2 最短时间的 Dijkstra 算法过程

Dijkstra 算法^[4]是具有代表性的最短路算法，用于搜索从某一固定起点到其余各点的最短路径，适合解决静态道路规划问题。本题中，由于道路速度不同的动态特点，我们将距离权值转化为时间权值，运用 Dijkstra 算法寻求某类发射装置从某一待机地域到发射点位的最短时间及其对应的路径。为了避免重复并保留每一步的计算信息，采用了标号算法。

以待机区域 D 到发射点位 F 最短时间的路径为例，具体算法如下^[5]：

Step1: n 个节点构成顶点集 $V = \{v_1, \dots, v_n\}$ ，以待机区域 D 作为起点 u_0 ，令 $l(u_0) = 0$ ，令 $l(v_F) = \infty$ ， $S_0 = \{u_0\}$ ， $i = 0$

Step2: 对每个 $v_F \in S_i (S_i = V \setminus S_{i-1})$ ，用

$$\min_{u \in S_i} \{l(v_F), l(v_F) + w(uv_F)\}$$

代替 $l(v_F)$ ， $w(uv)$ 表示顶点 u 和 v 之间边的权值。计算 $\min_{u \in S_i} \{l(v_F)\}$ ，把达到这个最小值得一个顶点记为 u_{i+1} ，令 $S_{i+1} = S_i \cup \{u_{i+1}\}$ 。

Step3: 若 $i = |V| - 1$ ，则停止；若 $i < |V| - 1$ ，则用 $i+1$ 代替，转 (b)。

最后从待机区域 D 到发射点位 F 的距离由最后一次标号给出 $l(v_F)$ 。

(1) 第一阶段的 Dijkstra 算法

分析由于三类发射装置都有相同的道路速度特性，所以根据此算法求解的到各发射点位时间排序应具有相似性，选取单台发射装置暴露时间较短的前 12 个发射点位作为从一个待机区域将要到达的发射点位。

(2) 第二阶段的 Dijkstra 算法

利用 Dijkstra 算法求出 6 个转载区域分别到第一波发射点与第二波发射点的距离，进而根据已知的车速，求出相应的时间。最后得到 24 个第一波发射点经转载区域至 36 个第二波发射点的任意一条路径的所用时间。具体流程如下：

$F_i (i = 1, 2, \dots, 24)$ 表示第一波发射点位， $Z_j (j = 1, 2, \dots, 6)$ 表示 6 个转载区域， $S_k (k = 1, 2, \dots, 36)$ 表示第二波发射点位。

Step1: 通过 Dijkstra 算法，分别求出 F_i 到 Z_j 的时间 $t(F_i, Z_j)$ ，以及 Z_j 到 S_k 的时间 $t(Z_j, S_k)$ ；

Step2: 设 $\min t(F_i, Z_j) = Q$ ，并使 $t(F_1, Z_1) = Q$ ，将 Q 与 $t(F_2, Z_1)$ 进行比较，若 $Q < t(F_2, Z_1)$ ，则保留该 Q 值；若 $Q > t(F_2, Z_1)$ ，则令 $Q = t(F_2, Z_1)$ ；

Step3: 依此进行计算比较，使 i 从 1 至 24； j 从 1 至 6； k 从 1 至 36；最后得到最短时间 $\min t(F_i, Z_j)$ 和 $\min t(Z_j, S_k)$ ；

Step4: 计算得到总最短时间 $\min t(F_i, Z_j) + \min t(Z_j, S_k)$ 。

4.3.3 0-1 规划模型

设第 k 台发射装置在第 n ($n = 1, 2$, 阶段的机动路径节点集为 $V_k^n = \{v_{k,1}, v_{k,2}, \dots, v_{k,i}\}$, $t_{v_i^k v_{i+1}^k}^n$ 表示第 k 台发射装置在第 n 阶段相邻两节点的时间, T_{node}^n 表示第 n 阶段经过所有节点所用时间, T_{wait}^n 表示第 n 阶段的等待暴露时间, T^n 表示第 n 阶段的暴露时间, $t_{k,0}^n$ 表示第 k 台发射装置在第 n 阶段的出发时刻, $S_{v_i^k}^n$ 表示第 k 台发射装置在第 n 阶段经过第 i 个节点的時刻。

对划分的 2 个阶段进行分析, 要使每个阶段的暴露时间最短, 经过各节点总时间应最短, 即:

$$\min T_{node}^n = \sum_k \sum_i t_{v_i^k v_{i+1}^k}^n \quad (4-1)$$

第 n 阶段第 k 台发射装置出发时刻根据在第 n 阶段单台暴露时间最长的发射装置算出:

$$t_{k,0}^n = \max_{k'} \sum_i t_{v_i^{k'} v_{i+1}^{k'}}^n - \sum_i t_{v_i^k v_{i+1}^k}^n \quad (4-2)$$

第 n 阶段第 k 台发射装置在第 i 个节点的時刻为:

$$S_{v_i^k}^n = t_{k,0}^n + t_{v_{i-1}^k v_i^k}^n \quad (4-3)$$

第 n 阶段的总暴露时间为:

$$T_{node}^n = T_{node}^n + T_{wait}^n \quad (4-4)$$

(1) 第一阶段的 0-1 规划

A,B,C 三类发射装置平均分配在两个待机区域, 每个待机区域分别有 3, 3, 6 辆。为了便于处理, 将 C 车分为 C_1, C_2 各有三辆。因此将发射装置分成了 A,B, C_1, C_2 四类, 由一个待机区域的 12 个发射点位可以构成矩阵 $P = \{p_{im}\}_{12 \times 4}$, 对矩阵 P 进行 0-1 规划。

其中, 矩阵权值为第 m 类车从待机区域到第 i 个发射点位的最短时间:

$$p_{im} = t_{im}^{(1)}, (\text{其中 } m = 1, 2, 3, 4; i = 1, 2, \dots, 12) \quad (4-2)$$

目标函数为:

$$\min T_{node}^{(1)} = \sum_k \sum_i t_{v_i^k v_{i+1}^k}^{(1)} \quad (4-3)$$

一个发射点位只能容纳一台发射装置, 且每类发射装置在一个待机区域有 3 台, 故约束条件可描述为:

$$\text{s.t.} \begin{cases} \sum_{i=1}^{12} K_{im} = 3 \\ \sum_{m=1}^4 K_{im} = 1 \end{cases} \quad (4-4)$$

其中, K_{im} 表示第 m 类车在发射点位 i 的停放情况:

$$K_{im} = \begin{cases} 0, & \text{第 } m \text{ 类车不去往 } n \text{ 发射点位} \\ 1, & \text{第 } m \text{ 类车去往 } n \text{ 发射点位} \end{cases} \quad (4-5)$$

(2) 第二阶段的 0-1 规划

综合考量三类节点之间的路径, 要使第一波发射点到转载区域再到第二波发射点的整体暴露时间最短, 在不考虑多台发射装置在转载区域等待暴露时间的情

况下，目标函数为：

$$\min T_{node}^{(2)} = \sum_k \sum_i t_{v_i^k v_{i+1}^k}^{(2)} \quad (4-6)$$

由Dijkstra算法求解得到每一台发射装置到可能的36个发射点位最短时间，建立一个 24×36 的时间矩阵 $C = \{c_{ki}\}_{24 \times 36}$ 。

其中，矩阵权值由第 k 台发射装置到第 i 个发射点位最短时间给定，即

$$c_{ki} = \min t_{ki} \quad (4-7)$$

采用0-1规划：

$$x_{ki} = \begin{cases} 1, & \text{第 } k \text{ 台发射装置前往第 } i \text{ 个发射位点} \\ 0, & \text{第 } k \text{ 台发射装置不前往第 } i \text{ 个发射位点} \end{cases} \quad (4-8)$$

一台发射装置只能对应一个发射位点，且36个发射位点不一定全部选取，故有：

$$\begin{cases} \sum_{i=1}^{36} x_{ki} \leq 1 \\ \sum_{k=1}^{24} x_{ki} = 1 \end{cases} \quad (4-9)$$

整合公式 (4-6)，(4-9) 可得

$$\begin{aligned} \min T_{node}^{(2)} &= \sum_k \sum_i t_{v_i^k v_{i+1}^k}^{(2)} x_{ki} \\ \text{s.t.} &\begin{cases} \sum_{i=1}^{36} x_{ki} \leq 1 \\ \sum_{k=1}^{24} x_{ki} = 1 \end{cases} \end{aligned}$$

4.3.4 0-1 规划修正

(1) 节点等待时间修正

节点等待时间分为两部分：单车道逆行的节点等待时间和超车的节点等待时间。

包含节点节点等待时间的暴露时间由排队论的排队输出结果给出，根据FCFS 排队规则求解伪代码如下：

```
%输入数据:24 个车的行驶路线，到达各个节点的时间，离开各节点的时间
%输出结果：修正后的到达各节点的时间以及离开个节点的时间
a=0
if a==0
    m=0;
    a=1;
    for i=1:1:24
        输入第 i 个车的行驶路线，到达各个节点的时间，离开各节点的时间
        for k=1:1:经过的节点数
            m=m+1;
            t(m,:)= [第 k 个点出发的时间，到达第 k+1 个点的时间，第 k 个点的序号，第 k+1 个点的序号， i];
        end
    end
end
```

```

按照各时间小区间发生时刻对 t 进行重新排序
for k=1:1:m-1
    if t(j,1)<t(k,2)&节点 t (j,3) 与 t(j,4)间为单行道
        if t(j,3)=t(k,4)&t (j,4) =t(k,3)
            修正晚到达的车到达节点的时间以及离开时间
            a=0
        else if t(j,2)<y(k,2)&t(j,3)=t(k,4)
            修正晚到的车到达各节点的时间以及从节点的出发时间
            a=0 并且返回 2
        end
    end
    if t (k,3) 为转载点
        if t (j, 3) =t (k,3) &t(j,1)-t(k,1)<10
            修正第 j 个车经过各节点的到达时间以及出发时间
            a=0 并且返回 2
        end
    end
end
end
end
end
end

```

(2) 转载区域等待时间的修正

设 $f_{ij}(X, T)$ 表示车辆 j 到达 i 转载区域的时间, 转载区域的集合 Z , 对于转载点 $i \in Z$, 其容量均为 2, $g_{ij}(X, T)$ 为车辆 j 在转载区域 i 的等待时间, 采用 $d_i(X, T)$ ($i \in J$) 表示在该路径下与节点 i 发生冲突的次数, $d_i(X, T)$ 的确定方法如下:

(a) 根据已有的路线方案 (X, T) , 针对所有车辆, 算出车辆到达转载区域 i 的时间;

(b) 可以确定车辆先后到来顺序的集合 $g_{ij}^*(X, T)$, 集合内部顺序按到达时间由小到大排列;

(c) 令 $k = 1$, $d_i = 0$;

(d) 可令 $\alpha_k = 1$, $k+1 \rightarrow s$;

(e) 若 $f_{ij}(X, T) \in [g_{ijk}(X, T), g_{ijk}^*(X, T)]$, 则 $\alpha_k \leftarrow \alpha_k + 1, s \leftarrow s + 1$;

(f) 重复步骤 (5), 使得 $s=m$;

(g) 若 $\alpha_k > M_i$, 则 $d_i + 1 \rightarrow d_i$;

(h) 令 $G \leftarrow G - \{j_i\}$; $k \leftarrow k + 1$;

$g_{ijk}(X, T) = g_{ijk}^*(X, T)$, 可算 $g_{ijk}^*(X, T)$;

(i) 重复步骤 (4) 到步骤 (7), 直至 $k=m$ 。

4.4 模型求解

4.4.1 第一阶段 Dijkstra 算法求解

(1) 对A类发射装置到发射点位的最短时间求解

分别以待机区域D1、D2为起点, 运用Matlab对到各发射点位 (F01~ F60)

的最短时间求解，并对求解结果按时间升序排序，前12个发射点位的结果如表4-2所示。

表4-2 A类发射装置所用时间对应前12个发射点位情况

待机区域	发射点位	时间/分钟	待机区域	发射点位	时间/分钟
D1	F43	105.1519	D2	F24	55.80217
	F58	147.0017		F25	91.75916
	F57	151.6406		F47	102.9052
	F42	155.3681		F46	107.1894
	F41	162.5556		F44	117.207
	F34	178.4404		F45	120.6336
	F35	180.9335		F3	133.8618
	F31	183.9422		F2	133.9793
	F32	187.4086		F1	134.0959
	F29	190.3586		F26	136.0844
	F30	190.3586		F49	141.6816
	F33	192.9294		F48	142.378

(2) 对B类发射装置到发射点位的最短时间求解

同对A类发射装置求解方法，得到按照时间升序排序的前12个发射点位如表4-3所示。

表4-3 B类发射装置所用时间对应前12个发射点位情况

待机区域	发射点位	时间/分钟	待机区域	发射点位	时间/分钟
D1	F43	90.13019	D2	F24	47.83043
	F58	126.0014		F25	78.65071
	F57	129.9776		F47	88.20443
	F42	133.1727		F46	91.87666
	F41	139.3334		F44	99.6311
	F34	151.896		F45	102.5682
	F35	154.0329		F3	114.0723
	F31	156.6119		F2	114.173
	F32	159.583		F1	114.273
	F29	161.3423		F26	116.6438
	F30	161.3423		F49	120.1351
	F33	164.3152		F48	120.732

(3) 对C类发射装置到发射点位的最短时间求解

同对A类发射装置求解方法，得到按照时间升序排序的前12个发射点位如表4-4所示。

表4-4 C类发射装置所用时间对应前12个发射点位情况

待机区域	发射点位	时间/分钟	待机区域	发射点位	时间/分钟
D1	F43	78.86391	D2	F24	41.85163
	F58	110.2513		F25	68.81937
	F57	113.7304		F47	77.17887
	F42	116.5261		F46	80.39208
	F41	121.9167		F44	86.65721
	F34	132.2509		F45	89.22717
	F35	134.1207		F3	99.39677
	F31	136.3773		F2	99.48485
	F32	138.9771		F1	99.57234

	F29	140.0356		F26	102.0633
	F30	140.0356		F49	104.3018
	F33	143.1177		F48	41.85163

从表 4-2,4-3,4-4 中可以看出, A、B、C 三类发射装置从待机区域到发射点位最短时间按时间升序的前 12 个发射点位是相同的, 这与问题分析时的猜想相符合, 说明对于每一个待机区域, 可以选取前 12 个发射点位作为第一个波次的发射点位, 发射点位对应的发射装置通过 0-1 规划求解。

4.4.2 0-1 规划求解

(1) 第一阶段的0-1规划求解

通过LINGO对0-1规划目标函数进行求解, 得到待机区域D1到12个发射点位的最短暴露时间:

$$\min T_1 = 1759.380\text{min}$$

待机区域D2到12个发射点位的最短暴露时间:

$$\min T_2 = 1222.213\text{min}$$

第一阶段的最短暴露时间:

$$\min T^1 = 2981.593\text{min}$$

相应的分配方案及机动方案如表4-5所示。

表4-5 第一波次的分配方案及机动方案

待机区域	发射点位	发射装置	路径
D1	F31	A01	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow Z4 \rightarrow J37 \rightarrow F31$
	F32	A02	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow Z4 \rightarrow J37 \rightarrow F32$
	F33	A03	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow Z4 \rightarrow J37 \rightarrow F33$
	F29	B01	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow J6 \rightarrow J36 \rightarrow F29$
	F30	B02	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow J6 \rightarrow J36 \rightarrow F30$
	F35	B03	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow J6 \rightarrow Z4 \rightarrow J38 \rightarrow F35$
	F43	C01	$D1 \rightarrow J11 \rightarrow J46 \rightarrow F43$
	F58	C02	$D1 \rightarrow Z03 \rightarrow J61 \rightarrow F58$
	F57	C03	$D1 \rightarrow Z03 \rightarrow J61 \rightarrow F57$
	F42	C04	$D1 \rightarrow J11 \rightarrow J46 \rightarrow J44 \rightarrow F42$
	F41	C05	$D1 \rightarrow J11 \rightarrow J46 \rightarrow J44 \rightarrow F41$
	F34	C06	$D1 \rightarrow J9 \rightarrow J8 \rightarrow J7 \rightarrow Z4 \rightarrow J38 \rightarrow F34$
D2	F1	A04	$D2 \rightarrow J12 \rightarrow J13 \rightarrow J21 \rightarrow F1$
	F26	A05	$D2 \rightarrow J32 \rightarrow J33 \rightarrow J34 \rightarrow F26$
	F50	A06	$D2 \rightarrow J3 \rightarrow J4 \rightarrow J50 \rightarrow F50$
	F3	B04	$D2 \rightarrow J12 \rightarrow J13 \rightarrow J21 \rightarrow F3$
	F2	B05	$D2 \rightarrow J12 \rightarrow J13 \rightarrow J21 \rightarrow F2$
	F49	B06	$D2 \rightarrow J3 \rightarrow J4 \rightarrow J5 \rightarrow J49 \rightarrow F49$
	F24	C07	$D2 \rightarrow J32 \rightarrow F24$
	F25	C08	$D2 \rightarrow J32 \rightarrow J33 \rightarrow F25$
	F46	C09	$D2 \rightarrow J3 \rightarrow J48 \rightarrow F46$
	F47	C10	$D2 \rightarrow J3 \rightarrow J48 \rightarrow F47$
	F44	C11	$D2 \rightarrow J3 \rightarrow J2 \rightarrow J47 \rightarrow F44$
	F45	C12	$D2 \rightarrow J3 \rightarrow J2 \rightarrow J47 \rightarrow F45$

(2) 第二阶段的0-1规划求解

利用LINGO对第二阶段的规划求解，得到第二阶段经过各节点的总时间：

$$\min T_{node}^{(2)} = 4603.5 \min$$

具体方案如表4-6，表4-7所示。

表4-6 从第一波发射点位到转载地域分配方案及机动方案

发射装置	发射点位	转载地域	路径
A01	F31	Z4	$F31 \rightarrow J37 \rightarrow Z4$
A02	F32	Z4	$F32 \rightarrow J37 \rightarrow Z4$
A03	F33	Z4	$F33 \rightarrow J37 \rightarrow Z4$
A04	F1	Z6	$F1 \rightarrow J21 \rightarrow J14 \rightarrow J15 \rightarrow J16 \rightarrow Z6$
A05	F26	Z2	$F26 \rightarrow J34 \rightarrow J5 \rightarrow J6 \rightarrow J51 \rightarrow Z2$
A06	F50	Z2	$F50 \rightarrow J50 \rightarrow J4 \rightarrow J5 \rightarrow J6 \rightarrow J51 \rightarrow Z2$
B01	F29	Z2	$F29 \rightarrow J36 \rightarrow J6 \rightarrow J51 \rightarrow Z2$
B02	F30	Z2	$F30 \rightarrow J36 \rightarrow J6 \rightarrow J51 \rightarrow Z2$
B03	F35	Z4	$F35 \rightarrow J38 \rightarrow Z4$
B04	F3	Z4	$F3 \rightarrow J21 \rightarrow J14 \rightarrow J15 \rightarrow J37 \rightarrow Z4$
B05	F2	Z6	$F2 \rightarrow J21 \rightarrow J14 \rightarrow J15 \rightarrow J16 \rightarrow Z6$
B06	F49	Z2	$F49 \rightarrow J49 \rightarrow J5 \rightarrow J6 \rightarrow J51 \rightarrow Z2$
C01	F43	Z5	$F43 \rightarrow J46 \rightarrow J44 \rightarrow Z5$
C02	F58	Z3	$F58 \rightarrow J61 \rightarrow Z3$
C03	F57	Z3	$F57 \rightarrow J61 \rightarrow Z3$
C04	F42	Z5	$F42 \rightarrow J44 \rightarrow Z5$
C05	F41	Z5	$F41 \rightarrow J44 \rightarrow Z5$
C06	F34	Z4	$F34 \rightarrow J38 \rightarrow Z4$
C07	F24	Z1	$F24 \rightarrow J32 \rightarrow J33 \rightarrow J4 \rightarrow Z1$
C08	F25	Z1	$F25 \rightarrow J33 \rightarrow J4 \rightarrow Z1$
C09	F46	Z1	$F46 \rightarrow J48 \rightarrow Z1$
C10	F47	Z1	$F47 \rightarrow J48 \rightarrow Z1$
C11	F44	Z1	$F44 \rightarrow J47 \rightarrow J48 \rightarrow Z1$
C12	F45	Z1	$F44 \rightarrow J47 \rightarrow J48 \rightarrow Z1$

表4-7从转载地域到第二波发射点位分配方案及机动方案

发射装置	转载地域	发射点位	路径
A01	Z4	F9	$Z4 \rightarrow J37 \rightarrow J15 \rightarrow J25 \rightarrow J24 \rightarrow F9$
A02	Z4	F7	$Z4 \rightarrow J37 \rightarrow J15 \rightarrow J25 \rightarrow J23 \rightarrow F7$
A03	Z4	F37	$Z4 \rightarrow J38 \rightarrow J42 \rightarrow J40 \rightarrow F37$
A04	Z6	F13	$Z6 \rightarrow J26 \rightarrow F13$
A05	Z2	F39	$Z2 \rightarrow J52 \rightarrow J8 \rightarrow J42 \rightarrow J40 \rightarrow F39$
A06	Z2	F56	$Z2 \rightarrow J54 \rightarrow J55 \rightarrow J53 \rightarrow J56 \rightarrow J60 \rightarrow F56$
B01	Z2	F59	$Z2 \rightarrow J54 \rightarrow J55 \rightarrow J59 \rightarrow J62 \rightarrow F59$
B02	Z2	F60	$Z2 \rightarrow J54 \rightarrow J55 \rightarrow J59 \rightarrow J62 \rightarrow F60$
B03	Z4	F36	$Z4 \rightarrow J37 \rightarrow J15 \rightarrow J16 \rightarrow J39 \rightarrow F36$
B04	Z4	F10	$Z4 \rightarrow J37 \rightarrow J15 \rightarrow J25 \rightarrow F10$
B05	Z6	F12	$Z6 \rightarrow J26 \rightarrow F12$
B06	Z2	F27	$Z2 \rightarrow J51 \rightarrow J6 \rightarrow J5 \rightarrow J34 \rightarrow J35 \rightarrow F27$
C01	Z5	F40	$Z5 \rightarrow J41 \rightarrow F40$
C02	Z3	F38	$Z3 \rightarrow J52 \rightarrow J8 \rightarrow J42 \rightarrow J40 \rightarrow F38$

C03	Z3	F54	$Z3 \rightarrow J57 \rightarrow J58 \rightarrow J59 \rightarrow F54$
C04	Z5	F23	$Z5 \rightarrow J41 \rightarrow J18 \rightarrow J19 \rightarrow J31 \rightarrow F23$
C05	Z5	F20	$Z5 \rightarrow J41 \rightarrow J18 \rightarrow J29 \rightarrow F20$
C06	Z4	F11	$Z4 \rightarrow J37 \rightarrow J15 \rightarrow J25 \rightarrow F11$
C07	Z1	F51	$Z1 \rightarrow J50 \rightarrow J53 \rightarrow F51$
C08	Z1	F52	$Z1 \rightarrow J50 \rightarrow J53 \rightarrow J56 \rightarrow F52$
C09	Z1	F53	$Z1 \rightarrow J50 \rightarrow J53 \rightarrow J56 \rightarrow F53$
C10	Z1	F55	$Z1 \rightarrow J50 \rightarrow J53 \rightarrow J56 \rightarrow J60 \rightarrow F55$
C11	Z1	F28	$Z1 \rightarrow J4 \rightarrow J5 \rightarrow J34 \rightarrow J35 \rightarrow F28$
C12	Z1	F48	$Z1 \rightarrow J4 \rightarrow J5 \rightarrow J49 \rightarrow F48$

4.4.3 第二阶段修正

经过修正后，可以得到在第一阶段无等待时间，暴露时间即为经过各节点的行驶时间，为2981.6min。在第二阶段，等待时间为2264.6min，总暴露时间为6868.1min。

4.5 结果分析

第一阶段为从待机区域D1、D2到第一波发射点位，其暴露时间为2981.6min。分配方案及机动方案在作战区域中的路线用图4-1反映。

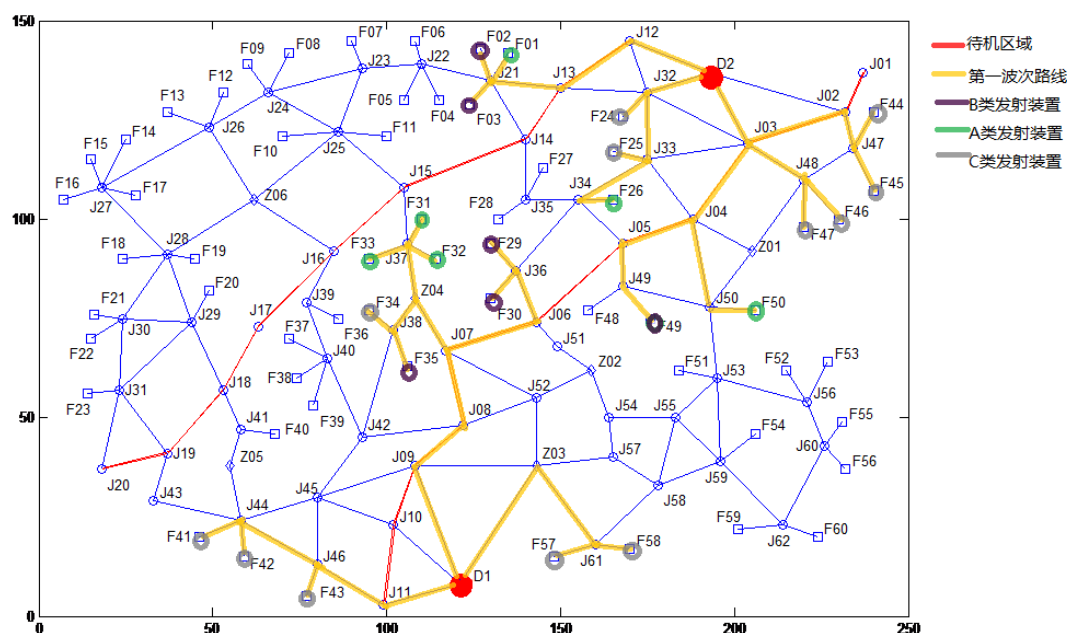


图4-1 第一波次到达发射点位路线图

第二阶段为从第一波发射点位到第二波发射点位，分配方案及机动方案在作战区域中的路线用图4-2，图4-3反映。

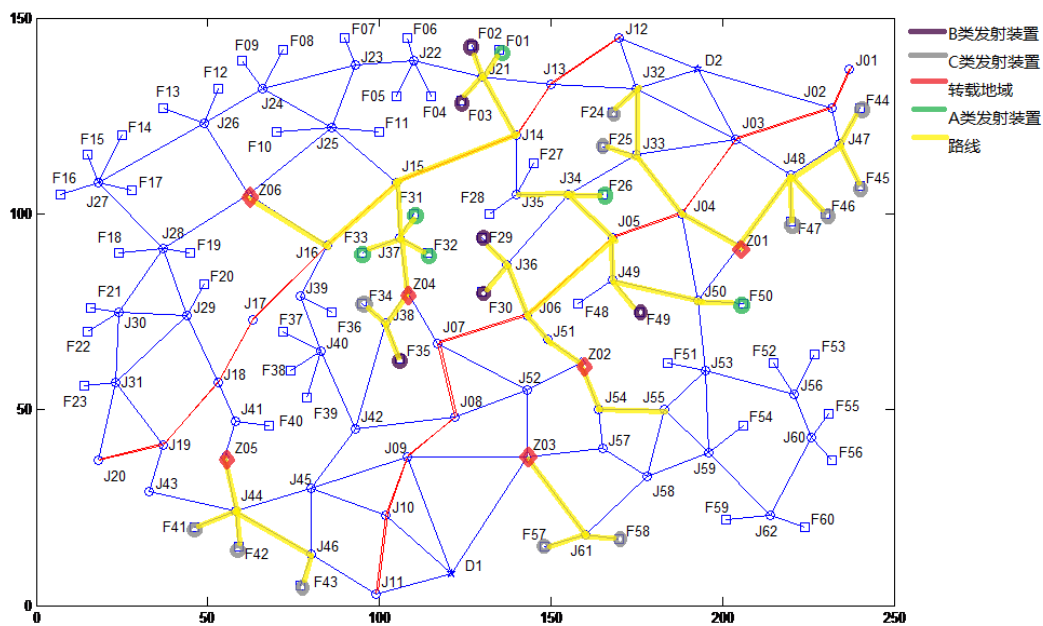


图4-2 从发射点位到转载地域路线图

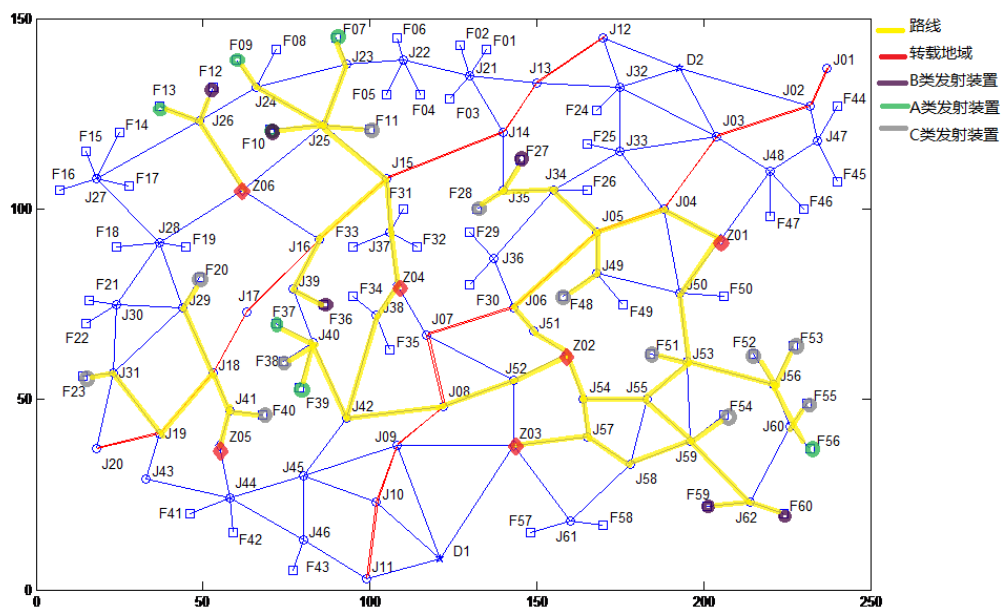


图4-3 从转载地域到第二波发射地域

两个波次导弹运输的总暴露时间为 $9849.7min$ 。从分析中可以看出，第一阶段无节点等待时间，暴露时间仅为经过各节点的总时间，第二阶段的暴露时间除经过各节点的总时间外，还包括多辆（大于2台）发射装置在非隐蔽区域的等待装弹时间以及装弹完毕后在单行道逆行交会的等待时间，总暴露时间增加了 $2264.6min$ ，不可忽略，所以对第二阶段的0-1规划进行修正是有必要的。

5. 问题二：临时转载地域的部署

5.1 问题分析

问题二是属于5个道路节点选取2个作为临时转载地域的选址问题，使得完成两个波次的整体暴露时间最短。增加临时转载地域，可以分担上一问中多台发射装置前往一个转载地域的数量，减少在转载地域的等待暴露时间以及节点等待时间，因此使暴露时间更短^[6]。

临时转载地域选址有五个待选地址，因此转载基地的选择共有10种方案，对每种选址方案，第一阶段（即从待机区域到第一波次发射位点）的暴露时间是固定的，需要规划第二阶段的路径。我们拟采用上一问中建立的第二阶段模型，模拟每种方案的路径选择，得到完成两个波次的整体暴露时间。最后通过排序比较，可以得到使整体暴露时间最短的临时转载地域选址方案。

5.2 模型准备

为了便于在MATLAB中表示各节点，我们将各节点标记为数字，其中可供临时转载地域选择的节点对应数字如表5-1所示。

表5-1 可供选择的节点对应的数字标记

J25	J34	J36	J42	J49
93	102	104	110	117

临时转载地域有10种选址方案，方案选址如表5-2所示。

表5-2 10种选址方案

方案1	方案2	方案3	方案4	方案5	方案6	方案7	方案8	方案9	方案10
J25	J25	J25	J25	J34	J34	J34	J36	J36	J42
J34	J36	J42	J49	J36	J42	J49	J42	J49	J49

5.3 模型建立及求解

5.3.1 某一方案的整体暴露时间

（1）第一阶段

第一阶段的路径规划同上一问，在此不多赘述。增加临时转载地域对第一阶段规划无影响，整体暴露时间为

$$T^1 = 5483.5 \text{ min}$$

（2）第二阶段

第二阶段的路径规划与上一问类似，增加了2个临时转载地域，其时间矩阵会因转载地域分布改变而发生改变（如图5-1）。

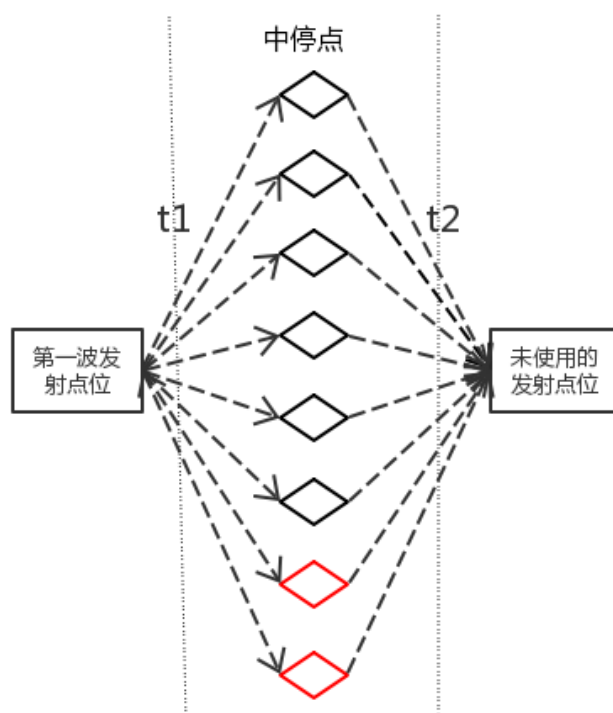


图5-1 第二阶段八个转载地域分析示意图

针对某一方案 A_i ($i=1,2,\dots,10$)，采用上一问最短时间的Dijkstra算法，分别设置第一波发射点位，转载地域为起止点和转载地域，未使用的发射点位为起止点，求得第二阶段每种可能路径的时间：

$$t(F_i, Z_j) + t(Z_i, S_i)$$

继而得到时间矩阵 $C_{A_i} = \{c_{ki}\}_{24 \times 36}$ ，具体算法过程同上一问，在此不多赘述。

暴露时间的目标函数为：

$$\min T_{node, A_i}^{(2)} = \sum_k \sum_i t_{v_i^k v_{i+1}^k}^{(2)} x_{ki} \quad (5-1)$$

约束条件同上一问：

$$\text{s.t.} \begin{cases} \sum_{i=1}^{24} x_{ki} \leq 1 \\ \sum_{j=1}^{36} x_{ki} = 1 \end{cases}$$

对10种不同方案，重复上述过程，得到暴露时间如表5-3所示。

表 5-3 增加临时转载地域后的整体暴露时间

选址方案	第二波暴露时间/分钟
方案1	5483.5
方案2	6846.8
方案3	6557.6
方案4	7065.1
方案5	5535.6
方案6	5483.5

方案7	5893.1
方案8	6492.8
方案9	6693.1
方案10	6991.4

5.3.1 选取最优方案

选取最优方案，具体流程如下：

输入:整数数组:Values

输出:Result

Result := -9999999;

for I:=0 to high(Values) do begin

Result := Min(Result,Values[I]);

end;

最后输出结果为方案 1，即最优方案为选取道路节点 J25,34 作为临时转载地域。

在该方案下，实施两个波次的发射任务运输规划如表 5-4 所示。

表 5-4 实施两个波次的发射任务运输方案

发射装置	待机区域	第一波发射点位	转载区域	第二波发射点位
A01	D1	F33	Z4	F60
A02	D1	F32	Z4	F36
A03	D1	F31	Z6	F16
A04	D2	F50	J34	F5
A05	D2	F26	J34	F22
A06	D2	F1	J25	F39
B01	D1	F35	Z4	F6
B02	D1	F30	Z4	F11
B03	D1	F29	Z2	F59
B04	D2	F49	J34	F10
B05	D2	F3	J25	F7
B06	D2	F2	J25	F9
C01	D1	F43	Z5	F23
C02	D1	F41	Z5	F40
C03	D1	F34	Z4	F38
C04	D1	F57	Z3	F54
C05	D1	F58	Z3	F51
C06	D1	F42	Z5	F20
C07	D2	F45	Z1	F48
C08	D2	F25	J34	F4
C09	D2	F24	J34	F28
C10	D2	F46	Z1	F52
C11	D2	F47	Z1	F53
C12	D2	F44	J34	F27

5.4 结果分析

我们选取暴露时间最短的方案 1，即选取道路节点 J25, J34 作为临时转载地域，转载地域布置如图 5-2 所示。在该方案的转载地域布置下，实施两个波次打击任务的整体暴露时间为 8465.09 min。

将表 5-4 的转载地域装弹情况和第一问情况进行对比，可以看出：

- ① 各个转载区域所停放的发射装置明显减少。
- ② 转载区域 Z2 所停放发射装置的数目变化最为明显，由原来的 5 辆变为 1 辆。

③ 转载区域 Z1 变化较为明显，原来停放在 Z1 的装置大部分停放在了临时转载地域。

所以增加临时转载地域可以很大程度上分担原转载地域装弹压力，从结果上看，整体暴露时间减少了 1385.6min。

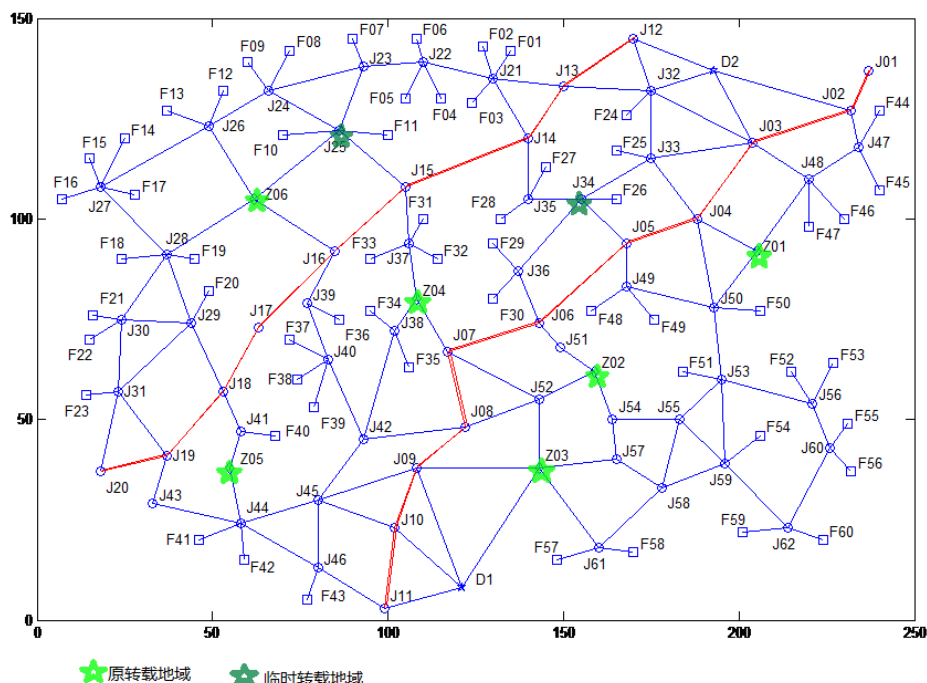


图5-2 增设临时转载地域布设图

6. 问题三：新增发射装置的隐蔽位点选取

6.1 问题分析

问题三是一个分配规划问题，在6个隐蔽位点中安排三台发射装置用于替代第一波次后的发射装置，完成第二波次的发射，要使第二阶段中暴露时间最短。6个隐蔽位点的容量为2台，所以可以将发射装置安排在3个隐蔽位点和2个隐蔽位点，总共有 $C_6^3 + A_6^2 = 50$ 种方案。我们拟将对每种方案进行分析，将暴露时间排序，得到最优方案。

首先，在第一波次完成后需要选出不参与第二波次的3台C类发射装置。基于第一问的结果，选择单台C类发射装置第二阶段暴露时间较长的3台遣返待机区域。然后针对新增发射装置的某方案暴露时间求解，在隐蔽位点的发射装置已经装弹，可以直接前往第二波发射位点，无中停点。参考第一问第二阶段的思想，采用时间的Dijkstra算法，固定起止点，求解到达剩余36个可能的发射点位的暴露时间。从而可以构建一个新增发射装置的 3×36 的时间矩阵，与其他参与第二波发射的21台发射装置结合构成一个新的 24×36 的时间矩阵。对该时间矩阵进行0-1规划，求解在该方案的情况下的最优发射方案及其暴露时间。最后，比较50种方案的暴露时间，得到使得暴露时间最短的新增发射装置隐蔽位点的最优方案。

6.2 模型准备

隐蔽位点的选取共有 50 种方案，具体如表 6-1 所示。

表 6-1 隐蔽位点选取方案

方案	隐蔽位点
方案 1	J04 J06 J08
方案 2	J04 J06 J13
方案 3	J04 J06 J14
方案 4	J04 J06 J15
方案 5	J04 J08 J13
方案 6	J04 J08 J14
方案 7	J04 J08 J15
方案 8	J04 J13 J14
方案 9	J04 J13 J15
方案 10	J04 J14 J15
方案 11	J06 J08 J13
方案 12	J06 J08 J14
方案 13	J06 J08 J15
方案 14	J06 J13 J14
方案 15	J06 J13 J15
方案 16	J06 J14 J15
方案 17	J08 J13 J14
方案 18	J08 J13 J15
方案 19	J08 J14 J15
方案 20	J13 J15 J15
方案 21	J04 J04 J06
方案 22	J04 J04 J08
方案 23	J04 J04 J13
方案 24	J04 J04 J14
方案 25	J04 J04 J15
方案 26	J06 J06 J04
方案 27	J06 J06 J08
方案 28	J06 J06 J13
方案 29	J06 J06 J14
方案 30	J06 J06 J15
方案 31	J08 J08 J04
方案 32	J08 J08 J06
方案 33	J08 J08 J13
方案 34	J08 J08 J14
方案 35	J08 J08 J15
方案 36	J13 J13 J04
方案 37	J13 J13 J06
方案 38	J13 J13 J08
方案 39	J13 J13 J14
方案 40	J13 J13 J15
方案 41	J14 J14 J04
方案 42	J14 J14 J06
方案 43	J14 J14 J08
方案 44	J14 J14 J13
方案 45	J14 J14 J15
方案 46	J15 J15 J04

方案 47	J15 J15 J06
方案 48	J15 J15 J08
方案 49	J15 J15 J13
方案 50	J15 J15 J14

6.3 模型建立

6.3.1 新增发射装置的时间矩阵

设某方案 Γ_i ($i=1,2,3,\dots,50$) 为三台发射装置的安置在 J_a, J_b, J_c 三个可供选择的隐蔽点位。

由于这三台发射装置不需要前往转载地域装弹,故运用时间的 Dijkstra 算法,设置起点分别为 J_a, J_b, J_c , 可以求解到 36 个发射点位的最短时间 $\min t_{J_i}$ (如图 6-1)。

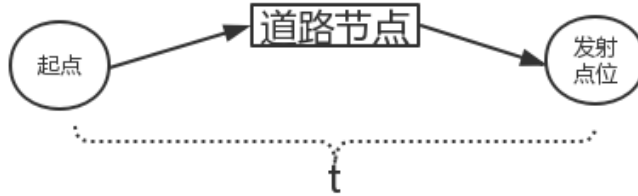


图 6-2 Dijkstra 算法起止点示意

以起点作为行, 终点作为列, 构建一个 3×36 时间矩阵 $Y = \{y_{ki}\}_{3 \times 36}$, 其中

$$y_{ki} = \min t_{J_i} \quad (6-1)$$

6.3.2 第二阶段 0-1 规划

选出不参与第二波次的发射装置后, 原矩阵 $C = \{c_{ki}\}_{24 \times 36}$ 转化为 $C' = \{c_{ki}\}_{21 \times 36}$, 结合新增加的发射装置的时间矩阵 $Y = \{y_{ki}\}_{3 \times 36}$, 构建一个新的 24 台发射装置的时间矩阵 $F = \{f_{ki}\}_{24 \times 36}$ 。

采用 0-1 规划, 暴露时间及目标函数为:

$$\min T_{node, F}^{(2)} = \sum_k \sum_i t_{v_i^k v_{i+1}^k}^{(2)} x_{ki} \quad (6-2)$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{24} x_{ki} \leq 1 \\ \sum_{j=1}^{36} x_{ki} = 1 \end{cases} \quad (6-3)$$

其中, x_{ki} 为 0-1 变量, 定义为

$$x_{ki} = \begin{cases} 1, & \text{第 } k \text{ 台发射装置前往第 } i \text{ 个发射位点} \\ 0, & \text{第 } k \text{ 台发射装置不前往第 } i \text{ 个发射位点} \end{cases} \quad (6-4)$$

6.3.3 确定最优隐蔽点位选取方案

最优方案的选取过程同第二问 5.3.1。

6.3 模型求解

6.3.1 第一波次后发射装置的筛除

基于第一问的结果，我们得到原 12 台 C 类发射装置第二阶段的暴露时间为如表 6-2 所示。

表 6-2 原 12 类 C 类发射装置第二阶段暴露时间

C 类发射装置	对应第一波发射点位	第二阶段暴露时间/分钟
C01	F43	193.5902
C02	F58	208.6689
C03	F57	257.9417
C04	F42	175.865
C05	F41	93.00728
C06	F34	168.8333
C07	F24	280.2398
C08	F25	244.1379
C09	F46	183.1029
C10	F47	166.3219
C11	F44	246.3039
C12	F45	249.7305

从表 6-1 中可以得到暴露时间前三的发射装置为 C07(280min)，C03 (258min),C12(250min)，所以选取这三台发射装置不参与第二波次导弹发射，而从发射位点遣返回待机区域。

6.3.2 新增发射装置后的暴露时间求解

表 6-3 50 种方案暴露时间

方案	第二波次暴露时间 /分钟	方案	第二波次暴露时间 /分钟
方案 1	5950.842	方案 26	5946.83
方案 2	5953.047	方案 27	5931.64
方案 3	5948.903	方案 28	5933.85
方案 4	5938.136	方案 29	5929.71
方案 5	5967.951	方案 30	5923.46
方案 6	5963.807	方案 31	5987.67
方案 7	5953.041	方案 32	5957.05
方案 8	5969.704	方案 33	5974.7
方案 9	5955.246	方案 34	5970.55
方案 10	5951.102	方案 35	5962.93
方案 11	5937.865	方案 36	5980.841
方案 12	5933.721	方案 37	5950.755
方案 13	5922.955	方案 38	5965.66
方案 14	5939.618	方案 39	5972.611
方案 15	5925.16	方案 40	5957.472
方案 16	5921.016	方案 41	5967.555
方案 17	5954.523	方案 42	5937.468
方案 18	5940.064	方案 43	5952.373
方案 19	5935.92	方案 44	5968.467
方案 20	5944.959	方案 45	5944.186
方案 21	5985.02	方案 46	5956.192
方案 22	5999.2	方案 47	5926.106

方案 23	6002.13	方案 48	5963.919
方案 24	6997.98	方案 49	5943.215
方案 25	5990.36	方案 50	5939.071

6.3.4 比较法确定确定隐蔽点位选取方案

通过表 5-3 的结果，选择暴露时间最短的方案，具体过程如下：

输入:整数数组:Values

输出:Result

Result := -9999999;

for I:=0 to high(Values) do begin

Result := Min(Result,Values[I]);

end;

最终得到结果为: 方案 16 暴露时间最短为 5921min, 即最优方案为选择 J06, J14, J15 作为新增发射装置隐蔽位点。

6.4 结果分析

经过分析，我们确定了选择J06, J14, J15为新增发射装置的隐蔽位点，隐蔽点位分布如图6-3。

在该情况下进行两波次的打击任务最短暴露时间为8902.6min，相比原最短暴露时间9849.7min，减少了947.1min。说明新增发射位点可以减少整体暴露时间。

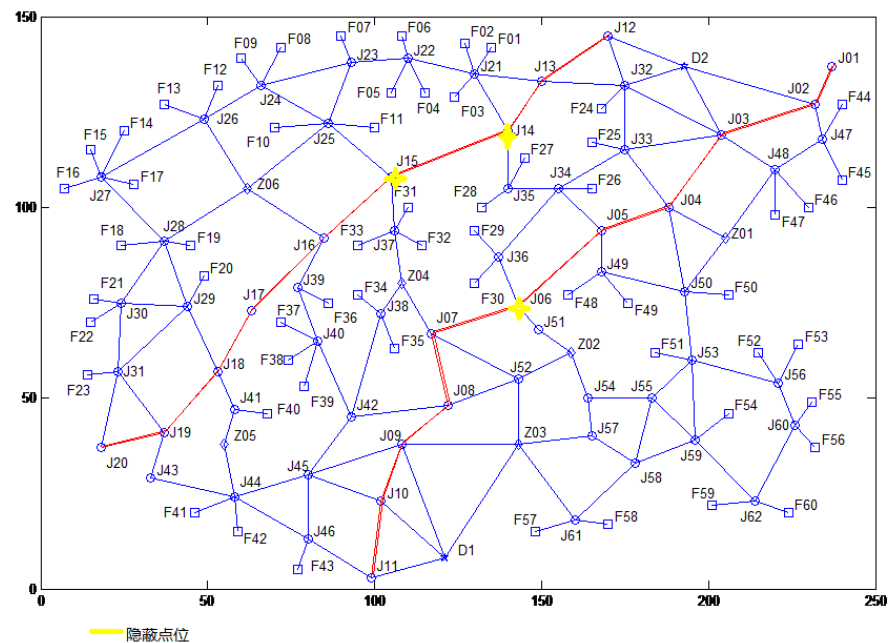


图6-3 隐蔽位点布设

7. 问题四：道路节点损毁的评价体系

7.1 问题分析

问题四需要建立一个道路节点损毁的评价体系。在敌我双方博弈中，容易被敌方攻击的节点是作战区域中的关键节点，节点的重要程度越高越容易被攻击。所以我们拟建立作战节点重要性评价体系，对作战区域的节点重要性进行定量分析，根据重要性指标确定最容易受到敌方攻击的3个道路节点。

评价体系由评价指标集和指标体系的递阶结构组成。根据军事作战体系网络节点特点，用度指标、介数指标、紧密度指标作为评价指标集，对每个指标进行量化分析和赋权。最后拟采用常用的TOPSIS综合评价方法进行综合评价，得到综合重要性排序。

7.2 三个评价指标的定义

三个评价指标定义如下^[7]：

(1) 度指标：表示在图论与网络理论中描述静态网络中节点对整个网络产生直接的影响大小；

(2) 紧密度指标：表示网络中某个节点到其它节点难易程度，可以用网络中某一节点到达其它所有节点距离之和的倒数来表示；

(3) 介数指标：表示某一节点在网络图中起枢纽作用的大小，可用各节点与其它节点最短路径经过该节点的次数来表示。

7.3 模型建立

7.3.1 三个指标归一化处理

设节点*i*的重要性为 I_i ，度指标、紧密度指标、介数指标分别用 $D_d(i)$ 、 $J_j(i)$ 、 $G_g(i)$ 表示，权值分别用 w_d 、 w_j 、 w_g 表示。则有：

$$I_i = w_d D_d(i) + w_j J_j(i) + w_g G_g(i) \quad (7-1)$$

对度指标进行归一化处理：

$$D_d(i) = D(i) / (n - 1) \quad (7-2)$$

其中D(i)表示各个节点的度，n表示节点个数。

对紧密度指标进行归一化处理：

$$J_j(i) = (n - 1) J(i) \quad (7-3)$$

其中

$$J(i) = \left[\sum_j^n d_{ij} \right]^{-1}$$

d_{ij} 表示节点*i*到节点*j*的最短距离。

介数指标进行归一化处理：

$$G_g(i) = 2G(i) / [(n - 1)(n - 2)] \quad (7-4)$$

其中

$$G(i) = \sum_{j < k} g_{jk}(i) / g_{jk}$$

7.3.2 TOPSIS 综合评价法

本问题可以采用 TOPSIS 综合评价法对节点重要性进行排序, 采用度指标、紧密度指以及介数指标作为评价指标, 基本原理如下:

(1) 根据归一化的指标建立属性决策问题的决策矩阵 $A = (a_{ij})_{mn}$, 规范化决策矩阵 $B = (b_{ij})_{mn}$, 其中,

$$b_{ij} = \frac{a_{ij}}{\sqrt{\sum_{i=1}^m a_{ij}^2}}, i = 1, 2, \dots, m; j = 1, 2, \dots, n \quad (7-5)$$

(2) 造加权规范阵 $C = (c_{ij})_{mn}$ 。设由决策人给定各属性的权重向量为 $w = [w_1, w_2, \dots, w_n]^T$, 则

$$c_{ij} = w_j b_{ij}, i = 1, 2, \dots, m; j = 1, 2, \dots, n \quad (7-6)$$

(3) 确定正理想解 C^* 和负理想解 C^0 。设正理想解 C^* 的第 j 个属性值为 C_j^* , 负理想解 C^0 第 j 个属性值为 C_j^0 , 则

$$\text{正理想解 } C_j^* = \begin{cases} \max c_{ij}, j \text{ 为效益型属性} \\ \min c_{ij}, j \text{ 为成本型属性} \end{cases} \quad j = 1, 2, \dots, n \quad (7-7)$$

$$\text{负理想解 } C_j^0 = \begin{cases} \min c_{ij}, j \text{ 为效益型属性} \\ \max c_{ij}, j \text{ 为成本型属性} \end{cases} \quad j = 1, 2, \dots, n \quad (7-8)$$

(4) 计算各方案到正理想解与负理想解的距离。备选方案 d_i 到正理想解的距离为

$$s_i^* = \sqrt{\sum_{j=1}^n (c_{ij} - c_j^*)^2} \quad i = 1, 2, \dots, m \quad (7-9)$$

备选方案 d_i 到负理想解的距离为

$$s_i^0 = \sqrt{\sum_{j=1}^n (c_{ij} - c_j^0)^2} \quad i = 1, 2, \dots, m \quad (7-10)$$

(5) 计算各方案的排序指标值, 即

$$f_i^* = \frac{s_i^0}{(s_i^0 + s_i^*)} \quad i = 1, 2, \dots, m \quad (7-11)$$

7.4 模型求解

三个指标的权值为 $w_d=0.25, w_j=0.25, w_g=0.5$ 。对62个道路节点进行评价, 列出重要性指数 $I_i \geq 0.4$ 的节点, 表7-1所示。

表7-1 综合评价评价结果

节点	重要性
J15	0.999989
J37	0.812289
J25	0.695476
J14	0.681972
J34	0.6158
J7	0.613774

J53	0.600945
J35	0.597569
J45	0.590142
J16	0.58474
J55	0.550304
J54	0.521945
J52	0.517218
J21	0.514517
J28	0.50844
J44	0.483457
J32	0.471303
J18	0.463876
J6	0.459824
J5	0.453747
J9	0.435517
J24	0.43214
J50	0.407833
J22	0.403106

7.4 结果分析

从权重中可以看出，介数指标对重要性的影响最大，当 $I_i \geq 0.4$ 时可以认为节点容易被损毁。结合作战区域图的分布，J25，J15，J37为相邻三个节点，虽然其重要性指数很大，但损毁一个节点会影响其他两个节点的重要性指数，所以最终我们认为J15，J55，J09这三个节点为敌方比较容易损毁的道路节点（如图7-1）。

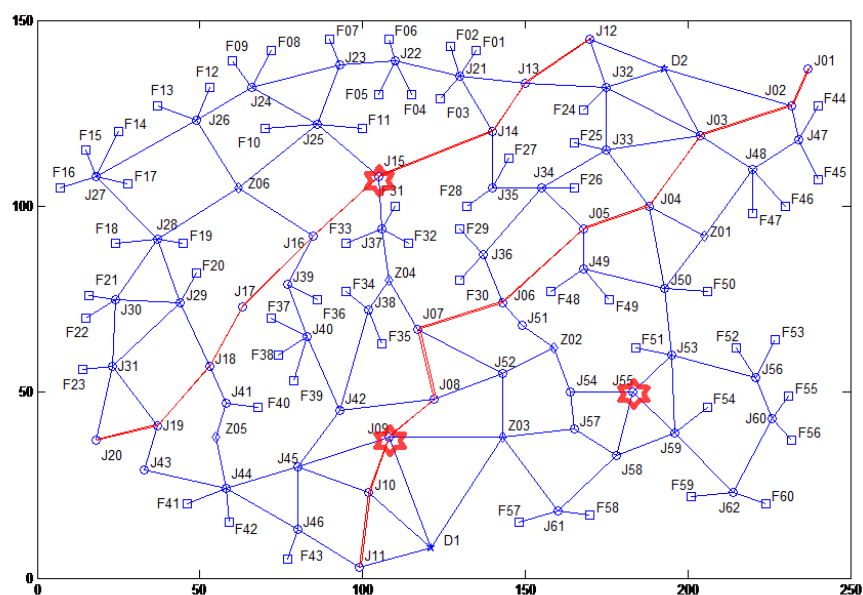


图7-1 损毁节点分布

8. 问题五：两个波次发射任务的优化

8.1 问题分析

问题五是要在保证整体暴露情况下,结合实际作战中规避敌方的侦察和打击等因素,对问题一的运输路径进行优化,以期提高运输的隐蔽性和可靠性^[8]。当前战争形式对导弹运输作战的要求是^[9]:在缩短整体暴露时间的同时,也要考虑道路的负荷能力,避免出现多个发射装置因堵塞而增大风险。所以,我们拟有选择的将运输任务尽量分散来提高隐蔽性和可靠性。在运输网络中,经常用运输流量的信息熵表示隐蔽性,我们拟建立一个以信息熵为目标函数的随机规划模型,用车辆运行时间和道路容量进行约束,得到优化的分散机动方案。

8.2 模型建立

记发射装置的道路方案为 (X, T) 。

其中, X 表示发射装置的运动路线, T 表示所用时间。

Step1: 首先考虑节点冲突

设 $S_{ij}(X, T)$ 为车辆 j 到达节点 i 的时间。

若存在 $k \neq j$, 有

$$S_{ij}(X, T) = S_{ik}(X, T) \quad (8-1)$$

则称节点 i 在这次实现中有冲突, 否则没有。

如果允许车辆在节点 i 停留, 设车辆 j 在节点 i 的停留时间为 t_{ij} 。

若存在 $k \neq j$, 有

$$[S_{ij}(X, T), S_{ij}(X, T) + t_{ij}] \cap [S_{ik}(X, T), S_{ik}(X, T) + t_{ik}] \neq \emptyset \quad (8-2)$$

则称节点 i 在这次实现中有冲突, 否则没有。

因此有

$$h_i(X, T) = \begin{cases} 1, & \text{节点 } i \text{ 不具有冲突性} \\ 0, & \text{节点 } i \text{ 具有冲突性} \end{cases} \quad (8-3)$$

得到期望优化目标

$$\max H \{h_i(X, T) = 1, i \in I\} \quad (8-4)$$

其中 I 为容量为 1 的节点的集合。

Step2: 目标的隐蔽性

提高隐蔽性就需要使任务分散到各条路线, 可以运用运输流量的信息熵来描述隐蔽性。

$$I(L_{ij}(X, T)) = - \sum_i \sum_{j \neq i} L_{ij}(X, T) \ln L_{ij}(X, T) \quad (8-5)$$

该式为 (X, T) 运输方案下运输网络的信息熵。

其中, L_{ij} 表示节点 i 和节点 j 间道路的运输流量。网络的信息熵越大, 说明整个任务的隐蔽性越好。期望优化目标:

$$\max - \sum_i \sum_{j \neq i} L_{ij}(X, T) \ln L_{ij}(X, T) \quad (8-6)$$

Step3: 发射装置运行时间

车辆 k 在节点 j 的到达时间:

$$S_{jk}(X, T) = t_k^0 + \sum_{i=1}^{j-1} \xi_{i,i+1}^{(k)} + \sum_{i=1}^{j-2} \eta_{i+1}^{(k)} (j = 2, 3, \dots, s_k) \quad (8-7)$$

其中, $\xi_{i,i+1}^{(k)}$ 表示发射装置从节点 i 到节点 $i+1$ 的随机行驶时间; $\eta_{i+1}^{(k)}$ 表示发射装置 k 在节点 $i+1$ 的停留时间。得到车辆 k 的总行驶时间为

$$S_k(X, T) = \sum_{j=1}^{s_k} [S_{(j+1)k}(X, T) - S_{jk}(X, T)] \quad (8-8)$$

$$S_{1k}(X, T) = t_k^0 \quad (8-9)$$

约束条件:

$$S_k(X, T) \leq T^*$$

其中, $k = 1, 2, \dots, m$ T^* 是总的任务时间。

Step4: 道路容量

节点 i 和节点 j 间道路的容量限制:

$$L_{ij}(X, T) \leq R_{ij} (i = 1, 2, \dots, n-1)$$

可建立随机规划模型, 约束条件为:

$$\max \begin{cases} \max H \{h_i(X, T) = 1, i \in I\} \\ \max H \{d_i(X, T) = 0, i \in J\} \\ \max - \sum_i \sum_{j \neq i} L_{ij}(X, T) \ln L_{ij}(X, T) \end{cases} \quad (8-10)$$

约束条件为:

$$\text{s.t.} \begin{cases} S_k(X, T) \leq T^* \\ L_{ij}(X, T) \leq R_{ij} \end{cases} \quad (8-11)$$

9. 模型评价及优化

9.1 模型优点及推广

1. 通过利用改进的 Dijkstra 算法, 可以简单的计算出任意两点间的时间。
2. 采用 0-1 规划模型, 采用较少的约束条件, 使问题更加简化。
3. 通过运用排队论的原理, 可以更加全面的考虑到发射装置在转载区域以及道路节点时的等待时间。对 0-1 规划所求得的整体暴露时间更贴近实际。
4. 利用 TOPSIS 综合评价方法, 可以更加客观真实的得到各个道路节点的综合重要性排序。

9.2 模型缺点及改进

1. 对最短时间的路径进行筛选时, 只是在方案个数不多的情况下可以实现, 若对于大数据方案, 则无法进行筛选。
 2. 采用 0-1 规划, 约束条件较少, 无法求得全局最优解。
- 模型的优化: 可以在传统筛选方法的基础上, 引入一种智能算法, 使之能够应对大数据, 从而进行智能筛选。

参考文献

- [1]李星毅,翟晓峰等.最小时间路径算法的改进及在路径优化中的应用[J].计算机应用研究,2008.6,25(6):1646-1647;
- [2]袁洪艳.基于排队论的医院全流程排队管理研究的研究[D].浙江大学.2008.6:11-18;
- [3]李元臣,刘伟群.基于 Dijkstra 算法的网络最短路径分析[J].微计算机应用,2004.5,25(3): 295-362;
- [4]汪民乐,高晓光.战略导弹机动阵地部署优化设计.系统工程理论与实践,1999.10(7): 94-98;
- [5]司守奎,孙兆亮.数学建模算法与应用[M].国防工业出版社,2015:38-40;
- [6]詹仁超,李应岐.基于混合算法的导弹部队铁路机动路径选择.四川兵工学报.2012.7(33):62-65;
- [7]李茂林,龙建国,张德群.基于复杂网络理论的作战体系节点重要性分析[J].指挥控制与仿真.2010.6,32(2): 15-27;
- [8]杨萍,刘卫东,李明雨.常规导弹波次作战中运输任务的规划[J].第三届不确定系统年会论文集.2005.8:284-289;
- [9]刘卫东.基于蚁群算法和禁忌搜索的部队机动路径选择[A].系统工程与电子技术.2008.8,30(8):1506-1508.

附录

附录 1: 任意相邻两点距离的 MATLAB 程序

```
x=shuju;  
for i=1:1:130  
    for j=1:1:130  
        a=[x(i,1) x(i,2)];  
        b=[x(j,1) x(j,2)];  
        c(i,j)=norm(a-b);  
    end  
end
```

附录 2: 计算邻接矩阵的 MATLAB 程序

```
x=shujuS1;  
linjiejuzhen=zeros(130,130);  
for i=1:1:349  
    m=x(i,1);  
    n=x(i,2);  
    linjiejuzhen(m,n)=c(i,3);  
end
```

附录 3: 三种类型车时间矩阵的 MATLAB 程序

```
AA=linjiejuzhen./45.*60;  
BB=linjiejuzhen./35.*60;  
CC=linjiejuzhen./30.*60;  
for i=69:1:78
```

```

        j=i+1;
AA(j,i)=linjiejuzhen(j,i)./70.*60;
BB(j,i)=linjiejuzhen(j,i)./60.*60;
CC(j,i)=linjiejuzhen(j,i)./50.*60;
end
for i=80:1:87
    j=i+1;
AA(j,i)=linjiejuzhen(j,i)./70.*60;
BB(j,i)=linjiejuzhen(j,i)./60.*60;
CC(j,i)=linjiejuzhen(j,i)./50.*60;
end
A1=AA;
A1(isinf(A1))=0;
[d1,p1,pred1]=graphshortestpath(sparse(A1),1,[9:68],'directed',false);
[d11,ind1]=sort(d1);
A2=BB;
A2(isinf(A2))=0;
[d2,p2,pred2]=graphshortestpath(sparse(A2),1,[9:68],'directed',false);
[d12,ind2]=sort(d2);
A3=CC;
A3(isinf(A3))=0;
[d3,p3,pred3]=graphshortestpath(sparse(A3),1,[9:68],'directed',false);
[d13,ind3]=sort(d3);
D(1,:)=d11;
D(2,:)=d12;
D(3,:)=d13;
paixu(1,:)=ind1;
paixu(2,:)=ind2;
paixu(3,:)=ind3;

```

附录 4：从出发到第一波齐射所用最短时间的 Lingo 程序

model:

```

sets:
Fashe/W1..W12/;
Leixing/J1..J4/;
links(Fashe,Leixing):c,x;
endsets
data:
c=
70.10125686    90.13018739    105.1518853    105.1518853
98.00111412 126.0014324    147.0016712    147.0016712
101.0937025    129.9776175    151.6405537    151.6405537
103.5787624    133.1726945    155.3681436    155.3681436
108.3703964    139.3333668    162.5555946    162.5555946
121.0660672    151.8960116 178.4403979    178.4403979
130.5501482    161.3422969    190.3585916    190.3585916
130.5501482    161.3422969    190.3585916    190.3585916
122.7281106 154.0329245    180.933463    180.933463
124.7339511 156.6118623 183.9422237    183.9422237
127.0448435    159.5830097    187.4085625    187.4085625
130.7254142    164.315172 192.9294185    192.9294185;

```

```

enddata
min=@sum(links:c*x);
@for(Leixing(j):@sum(Fashe(i):x(i,j))=3);
@for(Fashe(i):@sum(Leixing(j):x(i,j))=1);
@for(links:@bin(x));
End
附录 5: 从第一波导弹齐射结束后到第二波齐射就位的最短时间矩阵的 MATLAB
程序
[i,j,v]=find(CC);
b=sparse(i,j,v,130,130);
mm=[39 49 41 34 9 58 43 37 38 11 10 57 51 66 65 50 49 42 32 33 55 54 52 53];
nn=[4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 27 28 36 37 38 39 40
48 51 52 53 54 55 56 59 60];
for m=13:24 %固定发射点 1
for n=1:36%固定第二波发射点
for zz=3:8 %转载点
[x,y,z]=graphshortestpath(b,mm(m),zz,'Directed',false); %发射点到 Z
pp(zz-2)=x;
[x,y,z]=graphshortestpath(b,zz,nn(n)+8,'Directed',false); %Z 到第二波发射点
qq(zz-2)=x;
end
xmin=0;
for ii=1:6
vv(ii)=pp(ii)+qq(ii);
end %判断最小
xmin=min(min(vv));
xx(m,n)=xmin; %赋值
for ii=1:6
if vv(ii)==xmin cc(m,n)=ii;
end
end
end
end
end

```

附录 6: 从第一波导弹齐射结束到第二波导弹齐射的最短时间的 Lingo 程序
model:

```

sets:
Fashe/W1..W24/;
Leixing/J1..J36/;
links(Fashe,Leixing):c,x;
endsets
data:
c=157.2282077 157.2282077 151.9334413 130.9492165 142.8726315
139.6161524 118.8841462 116.2234119 128.245465 131.9791355
176.2297023 167.8608078 172.9087827 171.3038291 141.09752
134.4626573 160.8046569 161.9500281 164.927858 187.3112065
130.3343174 130.3343174 121.4792194 125.6766172 123.293396
126.43137 144.9764978 130.2899458 177.8961456 211.9001299
214.1160016 181.1429492 225.0911908 225.991233 214.3630004
210.8988693

```

217.7412509	217.7412509	212.4464845	191.4622597	203.3856747
200.1291956	179.3971893	176.7364551	182.5637322	
186.2974027	166.1890753	157.8201808	162.8681557	
161.2632021	131.056893	124.4220303	101.7379597	126.6089744
129.5868042	123.3279706	190.8473606	190.8473606	138.085125
157.8418937	155.4586726	158.5966466	62.00485374	
226.8356967	232.045631	266.0496153	268.265487	223.4784338
279.2406762	280.1407184	256.698485	253.2343539	
163.2196709	163.2196709	157.9249044	136.9406797	148.8640947
145.6076156	124.8756093	122.2148751	134.2369281	
137.9705986	182.2211655	173.852271	178.9002458	177.2952922
147.0889832	140.4541204	166.79612	167.9414912	170.9193211
193.3026696	136.3257806	136.3257806	127.4706826	
131.6680803	129.2848592	132.4228332	150.9679609	
136.2814089	183.8876087	217.8915931	220.1074648	
187.1344124	231.0826539	231.9826962	220.3544635	
216.8903325				
236.3994951	236.3994951	231.1047287	210.1205039	213.0286289
209.7721498	198.0554336	195.3946993	184.9640555	188.697726
232.9482929	224.5793983	229.6273732	228.0224196	
197.8161105	191.1812478	217.5232474	218.6686186	221.6464485
244.029797	177.2211264	177.2211264	200.6505069	204.8479046
202.4646835	205.6026575	201.6950883	152.1525081	
142.6281745	176.6321589	178.8480306	171.991979	189.8232197
190.723262	205.2120302	201.7478991		
222.8040924	222.8040924	217.5093259	196.5251012	194.6717498
191.4152707	184.4600308	181.7992965	166.6071764	
170.3408469	214.5914138	206.2225193	211.2704941	209.6655405
179.4592315	172.8243687	199.1663683	200.3117396	203.2895694
225.6729179	206.6548004	206.6548004	192.5810749	
201.9971001	199.613879	202.751853	183.8685305	206.6104288
204.8016167	238.805601	241.0214728	234.1654212	251.9966619
252.8967042	267.3854724	263.9213413		
212.5838535	212.5838535	207.2890871	231.7165593	251.3683163
248.1118372	227.3798309	224.7190967	259.7001498	263.4338203
307.6843872	299.3154927	304.3634675	302.7585139	
272.5522049	265.9173421	275.8051892	293.4047129	
296.3825428	297.3952001	140.2032089	140.2032089	
229.9749042	234.1723019	231.7890808	234.9270548	
261.8431501	115.1345906	105.6102571	139.6142414	141.8301131
135.0102728	152.8053023	153.7053445	168.2303239	
164.7661928				
199.571615	199.571615	192.7640582	165.7843406	181.1144456
176.9275439	150.2721073	146.8511633	208.1818907	212.9823241
269.8759101	259.1159029	265.6061563	263.5426445	224.7059614
216.1754236	205.0282472	237.005266	240.8339044	230.4779126
161.2182458	161.2182458	150.99511	159.0052843	155.9411428
159.9756808	192.2754191	158.9222736	223.3971449	
267.1165533	269.9655312	227.5716067	284.0764887	285.2336859
270.2831011	265.8292183			

261.0914616	261.0914616	254.2839048	227.3041872	242.6342922
238.4473905	211.7919539	208.3710099	269.7017373	274.5021707
331.3957567	320.6357495	327.1260029	325.0624911	286.225808
277.6952702	266.5480938	298.5251126	302.353751	291.9977592
213.687142	213.687142	212.5149566	220.5251309	217.4609894
221.4955274	252.3415751	181.4560614	176.8548693	
220.5742777	223.4232556	181.0293311	237.5342131	238.6914103
223.7408254	219.2869426			
261.0914616	261.0914616	254.2839048	227.3041872	242.6342922
238.4473905	211.7919539	208.3710099	269.7017373	274.5021707
331.3957567	320.6357495	327.1260029	325.0624911	286.225808
277.6952702	266.5480938	298.5251126	302.353751	291.9977592
213.687142	213.687142	212.5149566	220.5251309	217.4609894
221.4955274	252.3415751	181.4560614	176.8548693	
220.5742777	223.4232556	181.0293311	237.5342131	238.6914103
223.7408254	219.2869426			
280.5545691	280.5545691	273.7470123	246.7672947	244.3844144
240.1975127	231.2550614	227.8341173	208.3013915	213.1018249
269.9954109	259.2354037	265.7256571	263.6621453	
224.8254622	216.2949244	250.1632096	251.6358298	
255.4644682	284.2430591	257.947362	257.947362	241.6964038
255.7344004	252.670259	256.704797	224.9212044	255.6513898
260.9850429	304.7044513	307.5534292	297.6022805	
321.6643867	322.8215839	340.3137748	335.859892	
280.655236	280.655236	273.8476792	246.8679616	244.4850814
240.2981797	231.3557283	227.9347843	208.4020584	
213.2024919	270.0960779	259.3360706	265.826324	263.7628122
224.9261292	216.3955913	250.2638765	251.7364967	
255.5651351	284.343726	258.0480289	258.0480289	241.7970707
255.8350673	252.7709259	256.8054639	225.0218714	
255.7520567	261.0857098	304.8051182	307.6540961	297.7029474
321.7650536	322.9222508	340.4144417	335.9605589	
289.8437557	289.8437557	283.0361988	256.0564813	271.3865863
267.1996846	240.5442479	237.1233039	281.7243466	286.52478
343.418366	332.6583588	339.1486122	337.0851004	298.2484173
289.7178795	295.3003878	325.0587848	328.8874232	
320.7500532	215.5103651	215.5103651	241.2672506	
249.2774249	246.2132835	250.2478215	281.0938691	
183.2792845	173.1643875	216.883796	219.7327739	209.7816251
233.8437314	235.0009286	252.4931195	248.0392367	
360.8437071	360.8437071	352.9015574	321.4252203	339.3103428
334.4256241	303.3276148	299.3365134	310.7200075	
316.3205132	290.2668947	277.713553	285.2855152	282.8780848
237.5686212	227.6163271	193.5902212	230.8967433	
235.3634881	224.0357448	317.3314431	317.3314431	
244.0020967	277.7461222	274.1712906	278.8782516	
133.9905623	315.3841903	326.7765314	377.7825078	
381.1063154	315.4004199	397.5690991	398.9191625	
365.2304967	360.0343			
388.6272663	388.6272663	380.6851167	349.2087795	367.093902

362.2091834	331.111174	327.1200726	399.5263379	405.1268435
470.4306581	457.8773163	465.4492786	463.0418482	
417.7323846	407.7800905	373.7539846	411.0605066	415.5272514
383.3650363	303.1657194	303.1657194	304.060944	278.0662279
274.4913963	279.1983573	314.1543257	265.562792	228.480244
279.4862205	282.8100281	208.6689036	294.6902852	
296.0403486	258.4989803	253.3027837		
313.1201959	313.1201959	305.1780463	273.7017091	291.5868317
286.702113	255.6041036	251.6130022	262.5492748	268.1497805
242.096162	229.5428202	237.1147825	234.7073521	189.3978885
179.4455944	145.4194885	182.7260105	187.1927553	
175.8650121	269.6079319	269.6079319	195.831364	229.5753895
226.0005578	230.7075188	85.81982961	328.0139701	
339.4063111	390.4122876	393.7360952	328.0301997	410.1988789
411.5489423	377.8602764	372.6640798		
320.3076469	320.3076469	312.3654973	280.8891601	298.7742827
293.889564	262.7915546	258.8004532	269.7367258	275.3372315
249.283613	236.7302712	244.3022335	241.8948031	196.5853395
186.6330454	152.6069395	189.9134616	194.3802063	
183.0524631	276.7953829	276.7953829	203.018815	236.7628405
233.1880089	237.8949699	93.00728061	335.2014211	346.5937622
397.5997386	400.9235462	335.2176507	417.3863299	
418.7363933	385.0477274	379.8515308		
230.3404858	230.3404858	222.3983361	190.921999	208.8071215
203.9224028	172.8243934	168.8332921	241.2395573	246.840063
313.2159133	300.6625715	308.2345338	305.8271034	
260.5176398	250.5653457	239.1582014	276.4647234	
280.9314682	269.6037249	186.8282217	186.8282217	
174.5216465	183.0130999	179.4382682	184.1452292	224.279902
184.880969	259.0344648	310.0404413	313.3642488	263.9046703
329.8270326	331.177096	313.734747	308.5385504	
351.8025589	351.8025589	343.8604093	341.2215395	358.2936833
353.4089647	323.1239339	319.1328326	316.1968232	
321.7973289	388.1731793	375.6198375	383.1917998	
380.7843693	335.4749058	325.5226116	365.035611	366.7536679
371.2204127	404.7954354	243.2315921	243.2315921	324.821187
333.3126404	329.7378087	334.4447697	337.1844999	
205.6286646	193.1319311	244.1379076	247.4617152	237.2319547
263.9244989	265.2745623	287.0620314	281.8658348	
324.992524	324.992524	317.0503744	353.6915827	387.2004569
382.3157382	351.2177288	347.2266274	386.076859	391.6773647
458.053215	445.4998733	453.0718356	450.6644051	405.3549415
395.4026474	417.5515368	436.6337037	441.1004485	
447.9970603	216.4215571	216.4215571	352.9149819	
354.9977861	351.4229544	356.1299154	402.0877664	
178.8186297	166.3218962	217.3278727	220.6516803	
210.4219198	237.114464	238.4645274	260.2519965	255.0557999
329.2767953	329.2767953	321.3346456	357.9758539	391.4847281
386.6000095	355.5020001	351.5108987	390.3611303	395.9616359
462.3374863	449.7841445	457.3561068	454.9486764	

```

409.6392128    399.6869187    421.835808    440.9179749    445.3847197
452.2813315    220.7058284    220.7058284    357.1992531
359.2820573    355.7072257    360.4141867    406.3720376    183.102901
170.6061675    221.6121439    224.9359515    214.706191    241.3987352
242.7487986    264.5362677    259.3400711
354.8748627    354.8748627    346.932713    383.5739213    403.8360231
398.9513045    367.8532951    363.8621937    400.3786636
405.9791692    472.3550196    459.8016778    467.3736401
464.9662096    419.6567461    409.704452    434.187103    450.9355082
455.402253    464.6326265    246.3038958    246.3038958    369.5505481
378.0420015    374.4671699    379.1741309    416.3895709
208.7009684    196.2042349    247.2102113    250.5340189    240.3042584
266.9968026    268.346866    290.1343351    284.9381385
192.4533518    243.3311332    120.4370723    193.6750206    147.655839
57.40415302    57.28750998    57.17006523    101.5829189
101.5829189    93.64076923    130.2819775    179.3283254
174.4436067    143.3455973    139.3544959    211.7607612    217.3612669
283.7371172    271.1837754    221.0865496    209.6794053    68.54942562
150.3921415    150.3921415    106.5750062    110.0413449    194.8011059
288.1449442    188.6179024    192.0445229    178.6003691
174.3160978    159.0640437    227.4431275    278.449104
172.7718885    223.6496698    100.7556089    173.9935572    127.9743756
53.26016329    53.14352025    53.0260755    97.43892915    97.43892915
89.4967795    126.1379878    159.646862    154.7621433    123.664134
119.6730326    192.0792978    197.6798035    264.0556538    251.5023121
201.4050862    189.9979419    48.86796226    130.7106781
130.7106781    86.8935428    90.35988152    175.1196425    268.4634808
208.2993657    211.7259863    198.2818325    193.9975612    139.3825803
222.8197854    273.8257619
160.5837447    179.2496698    56.35560894    129.5935572    83.57437559
97.66016329    97.54352025    97.4260755    136.7802263    136.7802263
128.8380767    97.36173949    115.246862    110.3621433    79.26413397
75.27303258    147.6792978    153.2798035    219.6556538
207.1023121    157.0050862    145.5979419    93.26796226
168.7240064    168.7240064    42.4935428    45.95988152    130.7196425
224.0634808    252.6993657    256.1259863    242.6818325
238.3975612    183.7825803    267.2197854    318.2257619;
enddata
min=@sum(links:c*x);
@for(Leixing(j):@sum(Fashe(i):x(i,j))<=1);
@for(Fashe(i):@sum(Leixing(j):x(i,j))=1);
@for(links:@bin(x));
data:
@ole('D:\DATA.xlsx','ABC')=x;
enddata
end

```

附录 7：对从转载区域出发到第一波导弹齐射的最短时间进行修正的 MATLAB 程序：

```

clc;

```

```

clear;
load('SHU.mat');
load('AA.mat');
load('BB.mat');
load('CC.mat');
lj_dis = aa;
Lingjie_tA = AA;
Lingjie_tB = BB;
Lingjie_tC = CC;
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tA),1,[9:68],'directed',false);
T(1,:) = d1; T(2,:) = d1; T(3,:) = d1;
for k =1:60
    Lujing{1,k} = p1{1,k}; Lujing{2,k} = p1{1,k}; Lujing{3,k} = p1{1,k};
end
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tA),2,[9:68],'directed',false);
T(4,:) = d1; T(5,:) = d1; T(6,:) = d1;
for k =1:60
    Lujing{4,k} = p1{1,k}; Lujing{5,k} = p1{1,k}; Lujing{6,k} = p1{1,k};
end
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tB),1,[9:68],'directed',false);
T(7,:) = d1; T(8,:) = d1; T(9,:) = d1;
for k =1:60
    Lujing{7,k} = p1{1,k}; Lujing{8,k} = p1{1,k}; Lujing{9,k} = p1{1,k}; %
end
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tB),2,[9:68],'directed',false);
T(10,:) = d1; T(11,:) = d1; T(12,:) = d1;
for k =1:60
    Lujing{10,k} = p1{1,k}; Lujing{11,k} = p1{1,k}; Lujing{12,k} = p1{1,k};
end
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tC),1,[9:68],'directed',false);
T(13,:) = d1; T(14,:) = d1; T(15,:) = d1; T(16,:) = d1; T(17,:) = d1; T(18,:) = d1;
for k =1:60
    Lujing{13,k} = p1{1,k}; Lujing{14,k} = p1{1,k}; Lujing{15,k} = p1{1,k};
    Lujing{16,k} = p1{1,k}; Lujing{17,k} = p1{1,k}; Lujing{18,k} = p1{1,k};
end
[d1,p1,pred1]=graphshortestpath(sparse(Lingjie_tC),2,[9:68],'directed',false);
T(19,:) = d1; T(20,:) = d1; T(21,:) = d1; T(22,:) = d1; T(23,:) = d1; T(24,:) = d1;
for k =1:60
    Lujing{19,k} = p1{1,k}; Lujing{20,k} = p1{1,k}; Lujing{21,k} = p1{1,k};
    Lujing{22,k} = p1{1,k}; Lujing{23,k} = p1{1,k}; Lujing{24,k} = p1{1,k};
end
Fashe1 = xlsread('Diyibo.xlsx');
Fashe1 = reshape(Fashe1,60,24);

```

附录 8：对从第一波次结束到第二波次结束所用时间进行修正

```

clc;
clear;
load('SHU.mat');
load('AA.mat');
load('BB.mat');

```

```

load('CC.mat');
lj_dis = aa;
Lujing_tA = AA;
Lujing_tB = BB;
Lujing_tC = CC;
DanShuangdao = ones(130);
for t = 80:87
    DanShuangdao(t,t+1) = 0;
    DanShuangdao(t+1,t) = 0;
end
for t = 69:78
    DanShuangdao(t,t+1) = 0;
    DanShuangdao(t+1,t) = 0;
end

FS1 = [41 40 39 58 34 9 43 38 37 57 11 10 51 49 42 65 66 50 53 33 32 54 55 52];
Daixuan2 = setdiff(9:68,FS1);
for i = 1:6
    for j = 1:36
        mt = 1000;
        for k = 3:8

[Te_d1,Te_p1,Pre1]=graphshortestpath(sparse(Lujing_tA),FS1(i),k,'directed',false);

[Te_d2,Te_p2,Pre2]=graphshortestpath(sparse(Lujing_tA),k,Daixuan2(j),'directed',false);
        if Te_d1 + Te_d2 < mt
            mt = Te_d1 + Te_d2;
            T(i,j) = mt;
            Lujing2{i,j} = [Te_p1,Te_p2(2:end)];
            Zhuanzai(i,j) = k;
        end
    end
end
for i = 7:12
    for j = 1:36
        mt = 1000;
        for k = 3:8

[Te_d1,Te_p1,Pre1]=graphshortestpath(sparse(Lujing_tB),FS1(i),k,'directed',false);

[Te_d2,Te_p2,Pre2]=graphshortestpath(sparse(Lujing_tB),k,Daixuan2(j),'directed',false);
        if Te_d1 + Te_d2 < mt
            mt = Te_d1 + Te_d2;
            T(i,j) = mt;
            Lujing2{i,j} = [Te_p1,Te_p2(2:end)];
            Zhuanzai(i,j) = k;
        end
    end
end

```

```

        end
    end
end
for i = 13:24
    for j = 1:36
        mt = 10000;
        for k = 3:8

[Te_d1,Te_p1,Pre1]=graphshortestpath(sparse(Lujing_tC),FS1(i),k,'directed',false);

[Te_d2,Te_p2,Pre2]=graphshortestpath(sparse(Lujing_tC),k,Daixuan2(j),'directed',false);
        if Te_d1 + Te_d2 < mt
            mt = Te_d1 + Te_d2;
            T(i,j) = mt;
            Lujing2{i,j} = [Te_p1,Te_p2(2:end)];
            Zhuanzai(i,j) = k;
        end
    end
end
end
end

```

% 接下来运行 lingo 程序，将求出的 0-1 矩阵保存在 data.xlsx 中

```

Fashe2 = xlsread('Dierbo.xlsx');
Fashe2 = reshape(Fashe2,36,24);
Fashe2 = Fashe2';
for i = 1:24
    fs2(i) = find(Fashe2(i,:));
    Lujing2{i,1} = Lujing2{i,fs2(i)};
end
Lujing_tA_full = Lujing_tA + Lujing_tA';
Lujing_tB_full = Lujing_tB + Lujing_tB';
Lujing_tC_full = Lujing_tC + Lujing_tC';
for i = 1:24
    ll = Lujing2{i,1};
    Tm2_s(i,1) = 0;
    Tm2_e(i,1) = 0;
    if (i>=1 && i<= 6)
        for k = 2:length(ll)
            Tm2_e(i,k) = Tm2_s(i,k-1) + Lujing_tA_full(ll(k-1),ll(k));
            if ll(k)>=3 && ll(k)<=8
                Tm2_s(i,k) = Tm2_e(i,k) + 10;
            else
                Tm2_s(i,k) = Tm2_e(i,k);
            end
        end
    end
end
if (i>=7 && i <= 12)
    for k = 2:length(ll)

```

```

        Tm2_e(i,k) = Tm2_s(i,k-1) + Lujing_tB_full(ll(k-1),ll(k));
        Tm2_s(i,k) = Tm2_e(i,k);
        if ll(k)>=3 && ll(k)<=8
            Tm2_s(i,k) = Tm2_e(i,k) + 10;
        end
    end
end
if (i>=13 && i<= 24)
    for k = 2:length(ll)
        Tm2_e(i,k) = Tm2_s(i,k-1) + Lujing_tC_full(ll(k-1),ll(k));
        Tm2_s(i,k) = Tm2_e(i,k);
        if ll(k)>=3 && ll(k)<=8
            Tm2_s(i,k) = Tm2_e(i,k) + 10;    1
        end
    end
end
TM2_e{i,1} = Tm2_e(i,1:length(ll));
TM2_s{i,1} = Tm2_s(i,1:length(ll));
end

[kaoluT_chufa,kaoluT_daoda] =
correct_time(Lujing2,TM2_s,TM2_e,DanShuangdao)

```

附录 9：暴露时间修正的 MATLAB 函数

```

function[T_jiedianchufa,T_jiediandaoda]=correct_time(Lujing,TQ_jiedianchufa,TQ_j
iediandaoda,DanShuangdao)
T_jiedianchufa = TQ_jiedianchufa;
T_jiediandaoda = TQ_jiediandaoda;
tn = 0;
while tn < 10000
    tn = tn+1;
    m = 0;
    for i = 1:24
        ss = T_jiedianchufa{i,1};
        ee = T_jiediandaoda{i,1};
        ll = Lujing{i,1};
        for k = 1:length(ss)-1
            m = m + 1;
            T(m,:) = [ss(k),ee(k+1),ll(k),ll(k+1),i];
        end
    end
    [dd,in] = sort(T(:,1));
    for k = 1:m-1
        flag = 0;
        for j = k+1:m
            if T(in(j),1) < T(in(k),2) && DanShuangdao(T(in(j),3),T(in(j),4))
                if T(in(j),3) == T(in(k),4) && T(in(j),4) == T(in(k),3)
                    ss = T_jiedianchufa{T(in(j),5),1};
                    ee = T_jiediandaoda{T(in(j),5),1};
                    ll = Lujing{T(in(j),5),1};

```

```

    te = find(ll == T(in(j),3));
    ss(te) = T(in(k),2);
    ee(te+1) = ss(te) + T(in(j),2) - T(in(j),1);
    te = te + 1;
    while te<length(ss) && ss(te)<ee(te)
        u1 = ee(te) - ss(te);
        ss(te) = ee(te);
        ee(te+1) = ee(te+1) + u1;
        te = te + 1;
    end
    T_jiedianchufa{T(in(j),5),1} = ss;
    T_jiediandaoda{T(in(j),5),1} = ee;
    flag = 1;
    Finish = 0;
    break;
else if T(in(j),2) < T(in(k),2) && T(in(j),3) == T(in(k),3) &&
T(in(j),4) == T(in(k),4)
    ss = T_jiedianchufa{T(in(j),5),1};
    ee = T_jiediandaoda{T(in(j),5),1};
    ll = Lujing{T(in(j),5),1};
    te = find(ll == T(in(j),3));
    ee(te+1) = T(in(k),2);
    te = te + 1;
    while te<length(ss) && ss(te)<ee(te)
        u1 = ee(te) - ss(te);
        ss(te) = ee(te);
        ee(te+1) = ee(te+1) + u1;
        te = te + 1;
    end
    T_jiedianchufa{T(in(j),5),1} = ss;
    T_jiediandaoda{T(in(j),5),1} = ee;
    flag = 1;
    Finish = 0;
    break;
end
end
if T(in(k),3)>=3 && T(in(k),3)<=8
    if T(in(j),3) == T(in(k),3) && T(in(j),1) - T(in(k),1)<10
        ss = T_jiedianchufa{T(in(j),5),1};
        ee = T_jiediandaoda{T(in(j),5),1};
        ll = Lujing{T(in(j),5),1};
        te = find(ll == T(in(j),3));
        ss(te) = T(in(k),1) + 10;
        ee(te+1) = ss(te) + T(in(j),2) - T(in(j),1);
        te = te + 1;
        while te<length(ss) && ss(te)<ee(te)
            u1 = ee(te) - ss(te);
            ss(te) = ee(te);
            ee(te+1) = ee(te+1) + u1;

```

```

        te = te + 1;
    end
    T_jiedianchufa{T(in(j),5),1} = ss;
    T_jiediandaoda{T(in(j),5),1} = ee;
    flag = 1;
    Finish = 0;
    break;
end
end
end
if flag
    break;
end
end
end

```

附录 10：道路节点重要性评价的 MATLA 程序

%%求解各个节点的紧密度

A=linjiejuzhen;

m=[69:1:130];

for i=1:1:62

for j=1:1:130

A1(isinf(A))=0;

[d,p,pred]=graphshortestpath(sparse(A),m(i),[1:130],'Directed',false);

d=sum(d);

b(i,1)=1/d;

end

end

%%

n=63;

du=d./(n-1);%归一化度指标

[du1,ind1]=sort(du,'descend');

jinnmidu=(n-1).*jin;%归一化紧密度指标

[jinmidu1,ind2]=sort(jin,'descend');

jieshu=jie.*20./(n-1);

jieshu=jie./(n-2);%归一化介数指标

[jieshu1,ind3]=sort(jieshu,'descend');

%%求介数指标

aa=aa+aa';

N=size(aa,1);

B=zeros(1,N);

[D,C,aver_D]=Distance_F(aa); %C 是 ij 间最短路径条数

for k=1:N

for i=1:N

if i~=k%两端节点不算

for j=i+1:N %因为是无向的，所以正向、反向只算一次，所以只算一半；

%都算的话累加一起就是两倍了，也不影响

if j~=k %两端节点不算

if D(i,j)==D(i,k)+D(k,j)&C(i,j)~=0 %满足条件即证明 ij 间最短路径经过 k 节点

```

    B(k)=B(k)+C(i,k)*C(k,j)/C(i,j);
end
end
end
end
end
end

function [D,C,aver_D]=Distance_F(A)
%% 求复杂网络中两节点的距离、平均最短路径长度以及节点间最短路径条数
N=size(A,2);
D=A;
C=A;
C((C==inf))=0;
C((C~=0))=1;
D((D==0))=inf;
for i=1:N
    D(i,i)=0;
end
for k=1:N
    for i=1:N
        for j=1:N
            if D(i,j)>D(i,k)+D(k,j)
                D(i,j)=D(i,k)+D(k,j);
                C(i,j)=C(i,k)*C(k,j);
            elseif D(i,j)==D(i,k)+D(k,j)
                if k~=i&& k~=j
                    C(i,j)=C(i,j)+C(i,k)*C(k,j);
                end
            end
        end
    end
end
end
aver_D=sum(sum(D))/(N*(N-1));

```

附录 11： TOPISIS 评价模型的 MATLAB 程序

```

A=gezhibiaoS3;
W=[0.5 0.25 0.25];
%A 为决策矩阵, W 为权值矩阵,M 为正指标所在的列, N 为负指标所在的列
[ma,na]=size(A);          %ma 为 A 矩阵的行数, na 为 A 矩阵的列数
for i=1:na
    B(:,i)=A(:,i)*W(i);    %按列循环得到[加权标准化矩阵]
end
V1=zeros(1,na);           %初始化正理想解和负理想解
V2=zeros(1,na);
BMAX=max(B);               %取加权标准化矩阵每列的最大值和最小值
BMIN=min(B);               %
for i=1:na
    %if i<=size(M,2)       %循环得到理想解和负理想解, 注意判断, 不然

```

会超个数

```
V1(i)=BMAX(i);
V2(i)=BMIN(i);
%end
%if i<=size(N,2)
% V1(N(i))=BMIN(N(i));
% V2(N(i))=BMAX(N(i));
%end
end

for i=1:ma                                %按行循环求各方案的贴近度
    C1=B(i,:)-V1;
    S1(i)=norm(C1);                        %S1,S2 分别为离正理想点和负理想点的距离，用
二阶范数
    C2=B(i,:)-V2;
    S2(i)=norm(C2);
    T(i)=S2(i)/(S1(i)+S2(i));            %T 为贴近度
end
output_args=T;
[output,paixu]=sort(output_args,'descend');
```